Digital Pre-Distortion of Radio Frequency Digital to Analog Converters in a DOCSIS Application

A Thesis Submitted
to the College of Graduate Studies and Research
in Partial Fulfillment of the Requirements
for the Degree of Master of Science
in the Department of Electrical and Computer Engineering
University of Saskatchewan

by

Kristofer Dechka

Saskatoon, Saskatchewan, Canada

Permission to Use

In presenting this thesis in partial fulfillment of the requirements for a Postgraduate degree from the University of Saskatchewan, it is agreed that the Libraries of this University may make it freely available for inspection. Permission for copying of this thesis in any manner, in whole or in part, for scholarly purposes may be granted by the professors who supervised this thesis work or, in their absence, by the Head of the Department of Electrical and Computer Engineering or the Dean of the College of Graduate Studies and Research at the University of Saskatchewan. Any copying, publication, or use of this thesis, or parts thereof, for financial gain without the written permission of the author is strictly prohibited. Proper recognition shall be given to the author and to the University of Saskatchewan in any scholarly use which may be made of any material in this thesis.

Request for permission to copy or to make any other use of material in this thesis in whole or in part should be addressed to:

Head of the Department of Electrical and Computer Engineering
57 Campus Drive
University of Saskatchewan
Saskatoon, Saskatchewan, Canada
S7N 5A9

Acknowledgments

I would like to express my appreciation to my supervisor, Professor J. Eric Salt for his guidance and teaching.

Also I would also like to thank my wife Sara and my parents Steven and Diane Dechka for their patience and unwavering support.

Abstract

The use of Community Antenna Television Network (CATV) cable systems are a very common way that subscribers use to access the internet and download data. The transmitters that send the signals to subscribers must conform to a very stringent specification known as DOCSIS. Using traditional high frequency design techniques to meet this specification often lead to a lengthy and difficult production process where several calibrations have to be made.

In order to send a digitally modulated signal that conforms to the DOCSIS specification some sort of conversion between the discrete digital domain and the analog domain must occur. To accomplish this a Digital to Analog converter is used.

In recent years, the clocking or sampling frequency that can be used for Digital to Analog converters (DACs) has been rapidly increasing. The clocking frequency is directly proportional to the bandwidth that can be transmitted. DAC's that have exceptionally high clocking frequencies can be referred to as Radio Frequency DAC's. The clocking frequency of these devices has now progressed to the point where direct digital synthesis can be used for a DOCSIS transmitter without any analog frequency conversion stages.

Since Radio Frequency DAC's are real devices the output is not a perfect representation of the discrete signal that is sent to it. Unwanted distortion is added that can be measured at the analog output. Removal of this distortion or at least significantly reducing it could be the difference meeting or or not meeting the DOCSIS specification.

This thesis will explore the usage of these devices in this application. The basic structure of DAC's as well as the distortion signals themselves will be investigated in order to develop a method where the distortion can be removed. Ideally this can be done in a way that is suitable to be integrated into a transmitter architecture and meet the specification.

The frequency response of the major distortion products across the DOCSIS band is measured. Once this is done a way to match these frequency responses is needed so a cancellation signal can be created that removes the distortion. A method is developed that

uses an iterative algorithm to find filter coefficients whose frequency response matches that of the distortion signals as closely as possible. Since these cancellation signals are added to the discrete signal to be transmitted before the interface with the Radio Frequency DAC the process is known as pre-distortion.

The generated coefficients are used in digital filters as part of a pre-distortion design. Tests are performed with discrete signals that are close approximates to a DOCSIS signal that would be sent to a subscriber. Measured results show a decrease in the power of targeted distortion signals. The reduction of the distortion level is enough that the DOCSIS specification is met for all test signals.

Table of Contents

List of Tables

List of Figures

1. Introduction

1.1 Communication over Cable Television Networks

1.1.1 Data over Cable Evolution

Co-axial cable networks such as those possessed by Community Antenna Television (CATV) providers have several inherent advantages over other methods that provide broadband data services. Broadband is defined as "a transmission rate greater than the primary rate" (for example DS-1) by the International Telecommunications Union Standardization Sector (ITU-T). First of all, the cable is a center conductor shielded by a metal outer layer making the signals much less susceptible to ingress from surrounding sources like radio signals and spurious emissions from other transmitters [1]. There is also an extremely large band of spectrum available for data over cable service providers, from approximately 50MHz to slightly over 1000MHz.

The radio spectrum allocation for Canada shows there are many applications that compete for and have to share a similar bandwidth of wireless radio spectrum. This is illustrated in Appendix A. Clearly once the physical plant is in place the cable network has a great advantage.

CATV networks were initially designed to broadcast analog video one way with no return communication from the subscriber side. Two events played a large role in the widespread expansion and upgrade of cable networks. First, in the United States, the Telecommunications Act of 1996 led to the de-regulation of the telecommunications industry allowing CATV providers to provide services similar to those provided by telephone Local Exchange Carriers. [2] Second digital set top boxes were introduced in "1996 by both General Instru-

ment and Scientific Atlanta" [3]. These set top boxes differed from analog set top boxes in that they could demodulate signals that had been digitally modulated. In addition they could decode compressed digital video streams including MPEG-2 (Moving Picture Expert Group-2). The digital boxes could also digitally modulate and transmit signals on the return path to be received by the service provider.

The introduction of digital STB's allowed cable providers to start providing digital content such as internet access as well as allowing subscribers to combine several services into one package. For example, television, broadband internet access and telephone voice services could all now be obtained from the same provider and network. Since the CATV providers were now providing digital services along with existing analog television they came to be known as Multiple System Operators or MSO's

Cable TV networks began evolving into what became known as hybrid fibre/co-ax or HFC networks. HFC networks often have a SONET fibre-optic ring as the backbone with an opto-electrical transition from fibre nodes to co-axial cables for the final transmission to the subscribers premises [2]. A typical HFC system is similar to Figure 1.1

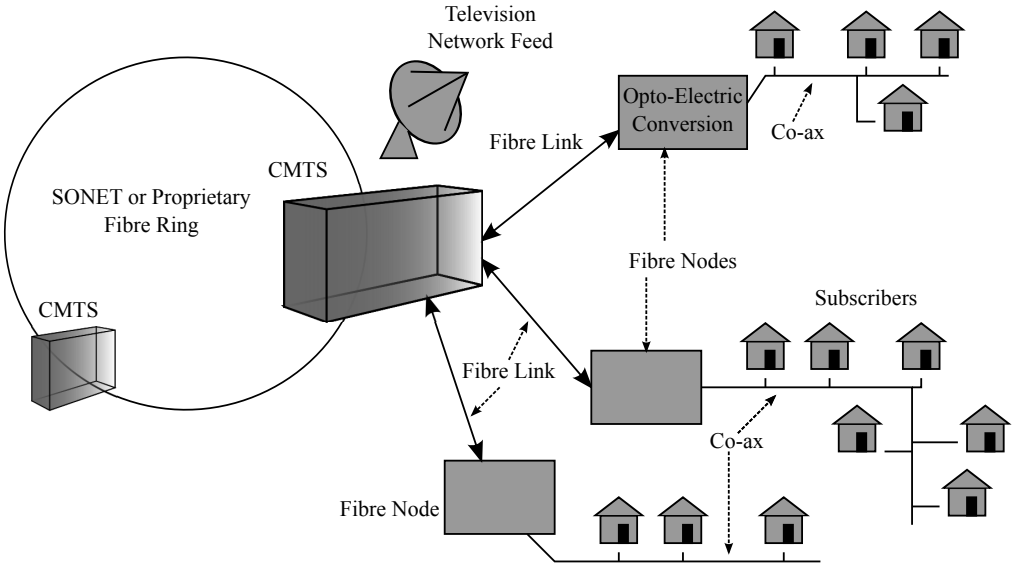


Figure 1.1: A Typical Hybrid Fiber Co-ax Network

1.2 Development of CATV Standards

1.2.1 CableLabs

At this time no system of standards existed for the allocation of bandwidth, signal type, power output, frequency allocation etc. that the service providers and digital STB's had to conform to. There was no real interoperability between various CATV networks and if a service provider wanted to expand they were generally limited to a single supplier for set top boxes and other equipment. If one service provider wanted to take over another network there was a real possibility that the two systems would not work together and the provider would either have to replace the hardware or make do with two completely different systems.

This was resolved when a partnership of some of the various service providers formed what was then known as the Multimedia Cable Network Systems Partners Ltd or MCNS [2](the predecessor of Cable Television Laboratories) to develop a standardized communication system. The main motivation for the service providers was to ensure that they were not limited to a single source for equipment and encourage competition between manufacturers resulting in lower equipment costs. One side effect of the standardization is that smaller companies and public domain researchers could now develop modular pieces that could be integrated into a larger CATV network. For example a smaller company could concentrate on just the manufacture of transmitters from a fibre node to subscribers or the receivers inside the set top boxes [4].

1.2.2 The DOCSIS Standard

The desire to provide data services and their inherent need for two-way communication led to the development of the Data Over Cable Service Interface Specification or DOCSIS. "This DOCSIS specification is the result of a cooperative effort undertaken at the direction of Cable Television Laboratories, Inc. for the benefit of the cable industry and its customers. [5]" CableLabs as Cable Television Laboratories is commonly known released DOCSIS 1.0 in 1997 [2]. Since that time the increased demand for data capacity (in both directions), new features such as Voice over Internet Protocol (VOIP) Video on Demand (VOD) and the

desire to improve the reliability of the signals has lead to the release of subsequent DOCSIS standards, the most recent being DOCSIS 3.0 released in 2006.

Upgrading the networks to provide broadband services and the the development of the DOCSIS standard led to a massive increase in the digital subscriber base. Table 1.1 shows the increase in subscriber numbers between 1997 and 2012 provided by the National Cable & Telecommunications Association (NCTA) [6].

Table 1.1: Subscribers for High Speed Internet over Cable

Year	Subscribers (Millions)
1997	0.1
1998	0.5
1999	1.5
2000	4.0
2001	7.3
2002	11.6
2003	16.5
2004	21.0
2005	25.4
2006	28.9
2007	35.7
2008	39.3
2009	41.8
2010	44.4
2011	47.3
2012	49.7

1.2.3 DOCSIS System Architecture

A DOCSIS system is essentially a point to multi-point communication system where the head-end Cable Modem Termination System or (CMTS) has two-way communication with multiple subscriber stations consisting of digital set top boxes that are commonly referred to as Cable Modems. The CMTS is responsible for allocating available bandwidth as well as monitoring and communicating with the various subscribers. While the latest DOCSIS releases have greatly increased the communication bandwidth from the subscribers to the head-end the majority of the bandwidth is used for data sent from the head-end to the subscribers. In a DOCSIS cable system the band will often be partitioned into upstream and downstream segments similarly to Figure 1.2. Downstream refers to signals sent from the CMTS to the cable modem while upstream refers to signals sent from the subscriber to the CMTS.

Contained within the DOCSIS specification is the detailed description of the downstream signal referred to as the Downstream Radio Frequency Interface or DRFI. From the DRFI specification "In the downstream direction, the cable system is assumed to have a pass band with a lower edge between 50 and 54 MHz and an upper edge that is implementation-dependent, but is typically in the range of 300 to 870 MHz. [7]" While this is typical for a system, the usual desire is for a downstream transmitter to operate from 54MHz to 1002MHz with the ability to be frequency agile meaning to be able to place a signal anywhere in that band while still meeting the DRFI specification. While the lower frequency range is usually reserved for upstream communication any DRFI compliant transmitter must still be able to operate down to the lower frequency limit. The reasoning for this is the upstream band might not be in constant use or a CMTS may not need to use all the allocated bandwidth for the upstream leaving the band available for more downstream content.

The DOCSIS specification was designed not only for digital signals but also to accommodate the legacy analog video signals the cable systems were originally designed to transmit. The most enduring feature of these analog video signals is the transmission bandwidth used by the CMTS. The 6MHz bandwidth used in the North American Standard is the same bandwidth allocated to a color television channel in 1953 by the National Television Sys-

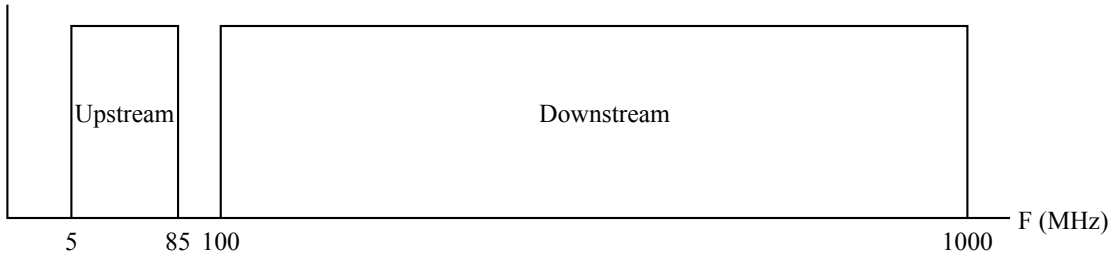


Figure 1.2: A typical frequency partitioning of signals in a DOCSIS network.

tems Committee (NTSC) of the Electronic Industries Association (EIA) [3]. Of course the 6MHz spectrum of an analog television signal, which includes separate carrier signals for the audio, video and color information, has a completely different spectrum than a digitally modulated signal, but the bandwidth used by a given channel is the same. The European 8MHz DOCSIS channel bandwidth was adopted from a similar analog video standard known as phase alternation line (PAL) that specifies various signal bandwidths up to a maximum of 8MHz.

1.2.4 DRFI

This sub-specification defines everything about the nature of the downstream signals that are sent on a cable network including signal format, symbol rate, interleaver depth, modulation type and the roll-off factor of the root raised cosine signal shaping filters. The specific reference is [ITU-T J.83-B] or Annex B for 6MHz North American channels and ETSI [EN 300 429] or Annex A for 8MHz European channels.

All of the various transmitter specifications contained in the DRFI are designed around the downstream channel bandwidths of 6MHz or 8MHz depending on which annex is being used. These bandwidths are referred to as simply the "channel bandwidth". Specifications including the allowed output power in an active channel as well as the spectral content of the unwanted or spurious emissions output by a transmitter are all measured over the channel bandwidth. The DRFI has been continuously updated since its original release in 2005, mainly due to CableLabs members finding problems or discrepancies with the specification. The current revision, I12 was released in November 2011 [7]. One very large difference

between I12 and previous releases is the expansion of the specification to allow a single transmitter to send more than 8 channels as well as specifying the spectral requirements to send non-contiguous active channels. With I12 the entire DOCSIS band can be filled with a single transmitter.

1.3 DOCSIS Headend Transmitter

While DRFI specifies the spectral components of the transmitter output, the design of the actual transmitter is dependent on the implementation and will depend on how a CMTS operator wants to connect to the router. Many early systems would have an analog interface to the transmitter with a single modulated signal provided as an input. This type of transmitter would behave much like a traditional radio and function mainly as a heterodyne frequency converter followed by a power amplifier. Later on digital interfaces to the transmitter such as ethernet or an MPEG packet stream would become far more common. Regardless of where it occurs, digital content must be converted to an analog signal that meets the DRFI specification. This conversion is performed by a device referred to as a digital to analog converter (DAC).

1.3.1 Transmitter Designs

There are several approaches to designing a DOCSIS headend transmitter. Transmitter designs can be categorized as direct conversion where then output of a single DAC covers the entire bandwidth specified by DRFI, or as frequency conversion where the outputs of several DAC's are translated in frequency and combined to cover the band. At present frequency conversion provides a higher quality signal since the performance of lower frequency DAC's is much better. With lower frequency DAC's the signal has to be converted into a narrow analog bandwidth and then translated in frequency with a technique known as mixing to place the signal at a desired center frequency. Both designs require power amplifiers as a final output stage.

DRFI compliant frequency translation, especially frequency translation where the signal can be placed anywhere over a large bandwidth is extremely challenging. Conventional

transmitters require multiple mixing stages using frequency agile oscillators. Oscillators are prone to spurious outputs, which are difficult to keep below the level required by the DRFI specification. Other problems include harmonic output, unwanted images of the desired signal [8], and issues caused by analog filtering that can distort the signal.

The result is that designers find it advantageous to build the broad band output spectrum with frequency agility in the digital domain [9] and translate the output with a single fixed local oscillator mixer. This greatly reduces the complexity of the analog circuitry, which is advantageous from a cost, repeatability, and reliability standpoint. Also processing the signal in the digital domain has several advantages over processing the signal in the analog domain.

1.4 Digital to Analog Converters

Modern digital to analog converters are integrated circuits that take a binary coded input and provide an analog output. The bandwidth of the analog output is directly related to the sampling frequency of the DAC. The Nyquist-Shannon sampling theorem in the original wording states "If a function x(t) contains no frequencies higher than B hertz, it is completely determined by giving its ordinates at a series of points spaced 1/(2B) seconds apart. [10]" In other words, to convert a signal from the analog to the digital domain it must be sampled at a frequency that is at least twice that of the bandwidth of the signal. The largest frequency that the analog signal can have and still satisfy the Nyquist-Shannon sampling theorum is traditionally known as the Nyquist frequency. The sampling clock determines the Nyquist frequency and therefore the maximum bandwidth of the analog signal. In order to accommodate the entire DRFI band the sampling clock would have to be in excess of 2004MHz.

1.4.1 RFDAC's

Improvements in the processes used to manufacture DAC's have progressed to the point where a DAC can accept a sampling clock input at a frequency traditionally used for the transmission of wireless signals. While there is no formal limit, once the frequency of the

sampling clock is above 1Gigahertz, a DAC is commonly referred to as a radio frequency DAC or an RFDAC[1]. The development of RFDAC's allow the direct digital synthesis of a DRFI compliant signal without any analog frequency translation. The increase in the configuration flexibility provided by RFDAC's allows many more channels to be broadcast while maintaining frequency agility [12]. In addition, the reduction of the number of the required analog components needed for a complete head-end transmitter make this a very exciting development. RFDAC's are currently available that have Nyquist frequencies in excess of 2.5GHz. The number of bits used in the digital interface can vary, but is commonly between 12 and 16 bits.

It is interesting to note that the maximum frequency of operation of an RFDAC is not usually limited by the sampling frequency, but by the rate at which the samples can be transferred to the RFDAC. Devices such as FPGA's or ASIC's that often provide the digital interface to RFDAC's become increasingly expensive as the digital connection frequency increases. High performance FPGA's have a maximum clocking frequency of approximately 700MHz [13]. For this reason techniques are employed such as double data rate (DDR) where data is clocked on both the positive and negative clock transitions as well as multiplexing where multiple samples are sent to the RFDAC in parallel. By using these techniques the transfer rate to the DAC is reduced to 1/4,1/8, or 1/16 etcetera.

1.4.2 Distortion in RFDAC's

The high operating frequency of RFDAC's stresses the performance of these devices. Inherent non-linearities and clocking glitches tend to be magnified as the operating frequency increases. The current technology, while steadily improving is such that the output of RFDAC's does not meet the DRFI specification. In particular there are distortion products related to the 2nd and 3rd harmonics of the desired output as well as distortion that appears to be from mixing with a continuous wave signal that has a frequency equal to the Nyquist frequency. Depending on the desired output frequency these spurious signals can violate the

[1]This term has also been used for a slightly different DAC architecture [11] and not just as a sampling frequency reference

DRFI specification. The frequency where this distortion appears is a function of both the signal frequency and the frequency of the sampling clock. Since output frequency agility within the DRFI bandwidth is a requirement, these spurious signals can not be removed with external filtering. Figure 3.1 shows a sinusoidal output from an RFDAC. The desired signal is identified by marker 1. The two main distortion products are identified by markers 2 and 3.

Many techniques have been investigated for eliminating the distortion. The most common technique uses pre-distortion in the digital domain so the final output meets specification. Pre-distortion involves non-linear digital signal processing to distort the signal in a way that cancels the distortion added by the RFDAC.

1.5 Problem Statement

RFDAC's can convert digital signals to analog signals at very high rates with bandwidths of a GHz or more. Unfortunately the conversion is not distortion free. Depending on the application the distortion may or may not prohibit the output of the RFDAC from meeting a specification, in this case the DRFI sub-specification contained in DOCSIS . The distortion can be significantly reduced by pre-distorting the digital signal to compensate for the distortion introduced by the RFDAC.

Manufacturers of RFDACs usually provide digital signal processing based pre-distorting compensation networks. These networks are integrated into the end users device that is connected to the RFDAC, such as an FPGA or ASIC. The compensation networks are provided as encrypted blocks of digital circuitry that are decoded when the software tools translate the hardware description language into an actual digital circuit. The networks are proprietary to the manufacturers and are made device specific, so they can not be used with any other RFDAC.

The problem with integrating a pre-defined signal processing block, like the compensation network, into a digital design is a loss of design freedom that can increase development cost and limit the ability to adapt the production environment to design changes. For example if it is necessary to change to a different RFDAC, for reasons of cost, reliability, or superior performance, the device hosting the compensation network must be re-programmed (FPGA) or re-fabricated (ASIC) and re-tested by the design engineers. In addition, the production engineers would have to redesign the system used on the factory floor to measure and calibrate different parameters used in the new pre-distortion network. In general algorithms used to calibrate compensation network parameters have to be changed, new test and measurement tools must be included in the production flow, and staff must be retrained to operate these tools.

For these reasons the goal of this M. Sc. project is twofold. First to develop a method that measures the distortion generated by an RFDAC, then develop an algorithm that uses the measurements to find parameters that pre-distort the digital signal to compensate for

this distortion in a way that is not specific to one manufacture's RFDAC.

2. Digital-to-Analog Converters

2.1 Functionality and Performance of ideal DACs

The function of a Digital-to-Analog Converter (DAC) is to convert an N-bit binary digital input, specifically an N-bit unsigned fraction, to a voltage that is the product of the unsigned fraction and a reference voltage. Normally the reference voltage is a constant and often generated internal to the DAC, but some DACs are designed so that the reference voltage can be varied from 0 volts to some maximum value. Variable reference voltage DACs can have an analog signal connected to the reference voltage terminal and produce an output that is the product of the digital and analog inputs. Such DACs are called multiplying DACs or MDACs.

One may wonder why DACs are designed for unsigned inputs when DSP engines used signed arithmetic and produce signed results. This is a non issue since the conversion from signed to unsigned numbers is trivial. For the purposes of explanation assume the signed output of the DSP engine is a signed fraction, which means it is restricted to the interval $[-1,\, 1 - 2^{-(N-1)}]$, where $N - 1$ is the number of fraction bits in the signed fraction. Adding 1 to the signed fraction and then dividing by 2 changes the interval to $[0,\, 1 - 2^{N}]$, which makes the result an unsigned number formatted for presentation to a DAC. The addition and division are accomplished in one simple operation, inverting the sign bit of the signed fraction. It turns out that regardless of whether or not the signed number is a signed fraction, the signed to unsigned conversion circuit is an inverter connected to the sign bit of the signed number.

The circuit for a DAC can be separated into a digital logic circuit and an analog circuit

that converts a binary valued input to an analog voltage. The basic structure for an N-bit DAC is shown in Figure 2.1. The digital circuit consists of a combinational logic circuit

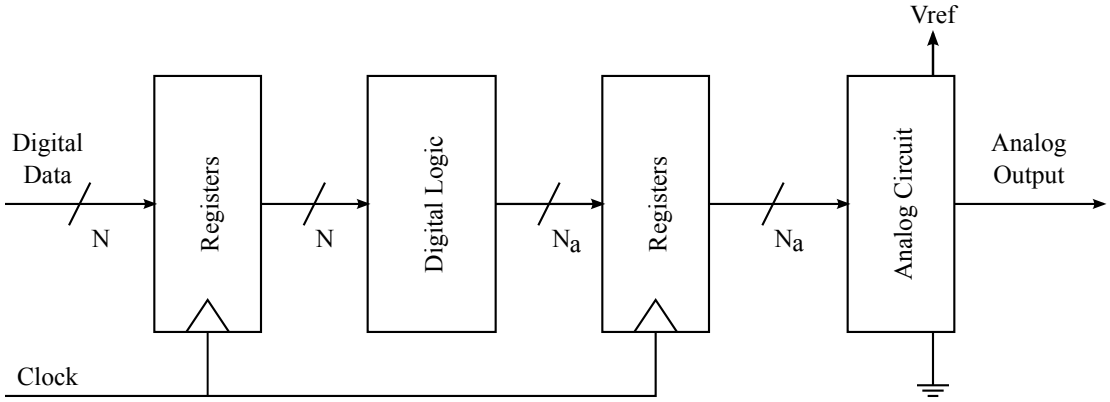


Figure 2.1: A block diagram of a generic DAC

between two sets of registers. The registers are clocked at the conversion rate. The combinational logic circuit converts the unsigned fraction input to a form more suitable for the analog circuit to convert to an analog voltage. Normally the number of bits output from the combination circuit is greater than N. The analog circuit has two inputs: a digital input (shown as N_a-bits in Figure 2.1) and a reference voltage input. The input can be a real or have both in-phase(I) and quadrature(Q) components [14]. This study does not consider multiplying DACs so the reference voltage is a constant.

The performance of an ideal DAC depends only on the number of bits used for the input. The output voltages are restricted to values $V_{ref} \times 2^{-N}(0, 1, \ldots, N-1)$, where V_{ref} is the reference voltage. The error in representing any real number in interval $[0, V_{ref}]$ lies in the interval $V_{ref}[-2^{-N}, 0]$ or in the interval $V_{ref}[-2^{-(N-1)}, 2^{-(N-1)}]$ depending on whether the closest lower unsigned fraction or closest unsigned fraction[1] is used. Assuming the error is uniformly distributed in the interval, the mean values are $-2^{-(N+1)}V_{ref}$ or $-2^{-(2N+1)}V_{ref}$ depending on whether the closest lower unsigned fraction or closest unsigned fraction[2] is

[1]There is one interval in the case of closest unsigned fraction assignment where error exceeds the stated bounds. The exception is when the real number is between $(1-2^{-N})V_{ref}$ and V_{ref}, as a number in that interval can not be rounded up.

[2]Not being able to round the number if it is between $(1-2^{-N})V_{ref}$ and V_{ref} creates a small mean value.

used. The variance of the error is $2^{-2N}/12$ in both cases.

Normally the DAC output is AC coupled to an amplifier and therefore the mean value, which is the DC value, is of no interest. It is the variance, which is the AC power in the error sequence, that measures the performance of an ideal DAC. The variance of the quantization error is often referred to as the power in the quantization noise, but it is not the total power in the noise as the mean contributes power. Strictly speaking, the variance should be referred to as the AC power in the quantization noise.

2.2 Distortion in Practical DACs

Most DACs are designed for use in a variety of applications and the manufacturers supply a broad range of specifications [15]. Some of the specifications apply to only a few of the applications. Some types of distortion are specified in different ways to suit the different applications.

For example non-linearity in the input-output characteristic has different ramifications in a control application than it does in a communications application. Communications engineers see non-linearity as something that generates spurs or distorts a signal while a controls engineer sees non-linearity as something that makes the small signal gain a function of operating point and affects stability. For that reason the communication engineer finds the spurious free dynamic range specification more useful than the integral non-linearity specification.

2.2.1 Differential and Integral Non-linearity

The analog components used in the analog circuitry of the DAC are imprecise and cause the input-output characteristic to deviate from the ideal straight line. Two types of deviations are specified. The differential non-linearity (DNL) is the maximum error between ideal and actual increment of 1 least significant bit. It is the worst case incremental error in step sizes of 1 LSB. DNL is measured by incrementing the input from 0 to full scale by one LSB and observing the change in analog output voltage for each increment. The DNL is the magnitude of the largest deviation from the ideal step size.

The integral non-linearity (INL) is the maximum deviation from a pseudo ideal straight line. The straight line is not exactly an ideal straight line, which is defined as the line that starts at the origin and has a slope of 1. The straight line used to measure the integral non-linearity is the line that starts at the origin and goes through the endpoint of the input-output characteristic. Normally the maximum deviation occurs somewhere near the midpoint.

These non-linearities are similar to those in amplifiers, and like the non-linearities in amplifiers, cause spurs. Such non-linearities are typically modelled using a third order polynomial for the input-output characteristic, equation 2.1 is an example of this.

$$y(t)\big|_{nT_s \le t \le (n+1)T_s} = x[n] + \alpha_2(x[n] - 0.5)^2 + \alpha_3(x[n] - 0.5)^3 \tag{2.1}$$

α_2 and α_3 are small coefficients that fit the polynomial to the actual input-output characteristic. When $x[n]$ is a sinusoid the squared and cubed terms generate second and third harmonics.

Spurs caused by differential and integral non-linearities are often problematic in communications systems. However, the magnitude of the spurs can not be predicted from the specifications of DNL and INL. Another specification, referred to as the spurious free dynamic range, does that.

2.2.2 Glitch Impulses

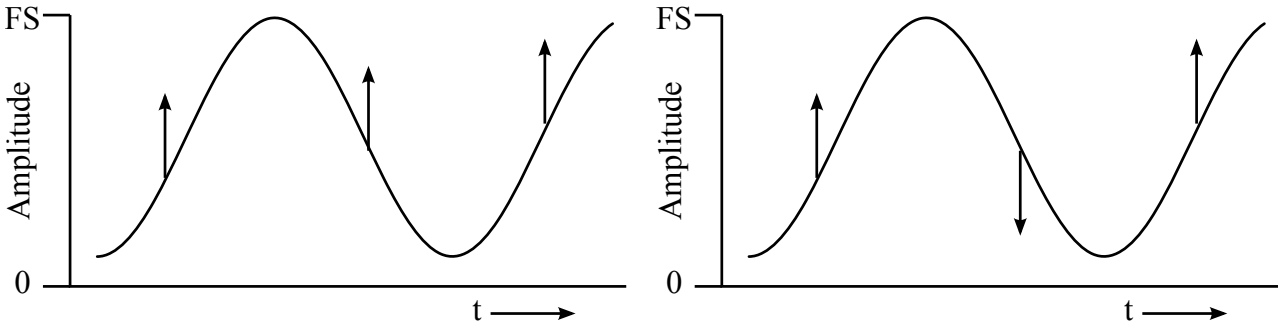
Each new output of a practical DAC starts with a transient. This transient is often under damped so is better characterized by its settling time than its rise time. The worst case settling time is usually given in the data sheet.

On top of the transient is an impulse like glitch caused by differences in propagation delays in the N_a digital signals that drive the analog circuit (see Figure 2.1). These glitches, although completely dissimilar to non-linearities, cause the same sort of spurs. The maximum area that a glitch may have is usually specified in the data sheet. The area of the glitch has units volt-seconds, which could be specified as least-significant-bits $\times$ minimum sampling-period. This worst case area is usually less than 1 LSB$\times$minimum-sampling-period. Again, while this sort of distortion causes spurs and is problematic to communications systems, the

size of the spurs can not be predicted from the glitch area specification.

Since the problem is to pre-distort the signal to suppress the spurs, it is important to understand the principles that cause the glitches. The mechanism depends on the format of the binary signal presented to the analog circuit in Figure 2.1. The principle will be explained for the case where the combinational circuit in Figure 2.1 does nothing and the N-bit input is presented to the analog circuit. N is taken to be three and two companion transitions are considered. The transitions are from 011 to 100, which is just under half scale to half scale and the companion transition from 100 to 011.

There are two phenomena that create glitches. One is a slight difference between low-high and high-low transition times and the other is a slight difference among the propagation times of the N bits. Suppose the low-high transition is faster than the high-low transition. Then for each new sample there is a brief intermediate state which introduces a glitch. For the first transition under consideration the initial, glitch and final states are 011, 111 and 100. The glitch has height 111-100=011. The companion transition has initial, glitch and final states 100, 111 and 011. In this case the glitch has height 111-011=100. While one transition represents a positive slope and the other a negative slope, both generate positive glitches. These glitches are shown superimposed on a sinusoid in Figure 2.2a.



(a) Low-high vs. High-low Transition Glitches (b) Differences in bit propagation times

Figure 2.2: Midscale glitches in a sinusoid

These glitches tend to cause second harmonics.

The second phenomenon has the propagation delays dependent on the bit positions. The

principle is explained under the assumption that the propagation delay of the most significant bit is smaller than the other two. The first transition has states 011, 111 and 100 making the glitch 011. The second transition has states 100, 000 and 011 making the glitch -011, which of course is negative. In this case the positive slope yields a positive glitch and the negative slope yields a negative glitch. These glitches are shown superimposed on a sinusoid in Figure 2.2b. These tend to cause third harmonics.

The larger glitches occur near 1/2 full scale. There are also significant glitches at 1/4 and 3/4 full scale, which are about half the height and one quarter the power of the glitches at 1/2 full scale. The glitches at 1/8, 3/8, 5/8 and 7/8 full scale are about one quarter the height and one sixteenth the power of the glitches at half scale. The glitches at half full scale are dominant in most communications systems.

2.2.3 Asymmetry in Dual Data Rate Clock

Asymmetry in a DDR clock that loads even samples on the positive edge and odd samples on the negative edge also causes spurs. This asymmetry is caused by the clock generating circuit which is external to the DAC so is not covered by any specification in the DAC data sheet. The asymmetry is analogous to the sampling clock having a duty cycle that is not exactly 50% and causes the analog output for even samples to be held slightly longer or slightly shorter than the odd samples. The same effect is obtained by taking a perfectly balanced output and multiplying the even samples by $(1+\epsilon)$ and the odd samples by $1-\epsilon$, where ϵ is a small constant.

This is the same as adding $\epsilon x[n]\cos(\pi n)$ to $x[n]$ prior to sending $x[n]$ to the DAC. If $x[n]$ is the sinusoid $\cos(\omega_o n)$, then asymmetry in the DDR clock causes a spur at frequency $\pi-\omega_o$ with amplitude ϵ.

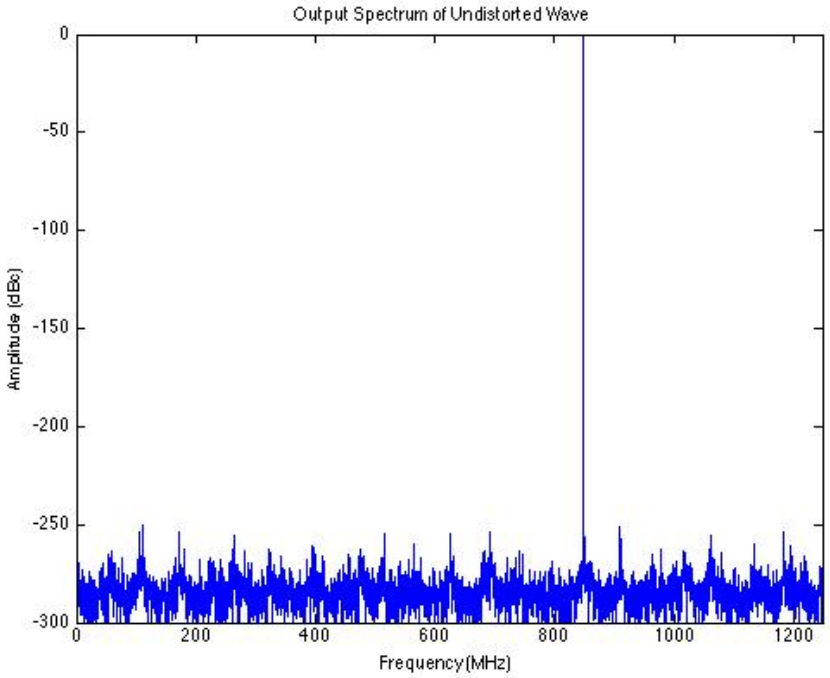
Modelling Spur creation

This method of spur creation can be tested in Matlab by multiplying the distortion factor with an expected signal. If the spurs match what is seen on a spectrum analyzer there might be some way of using the technique as a pre-distortion method. A test tone is generated at

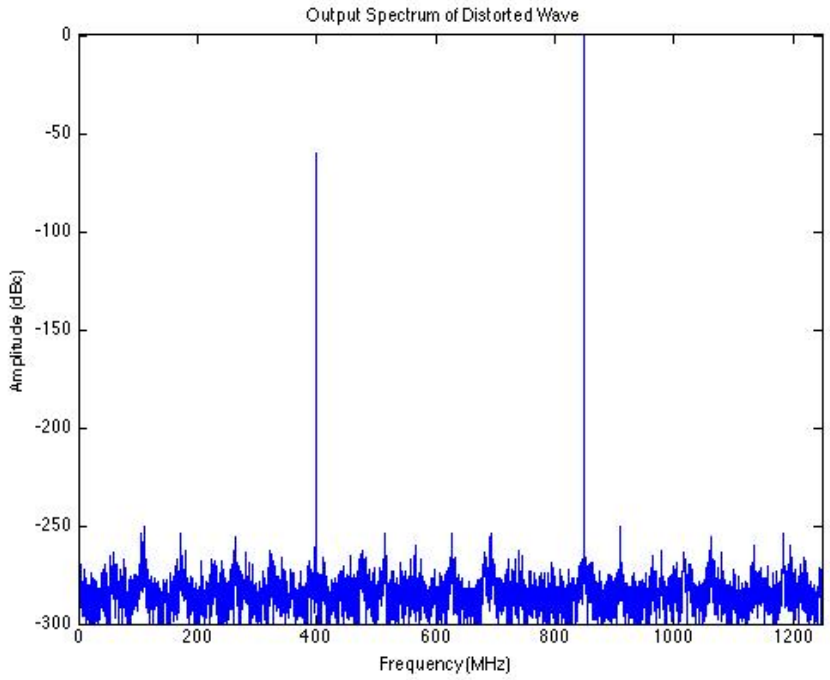
approximately 850MHz using the $1 \pm \epsilon$ multiplying factor, where ϵ is a value representative of the asymmetry in the sampling clock duty cycle.

Frequency plots of a regular sinusoid and the distorted version using an ϵ of 0.001 is shown in Figure 2.3 clearly there is a spur at $F_s/2 - F_c$. The spur level is close to $20 \times \log_{10}(0.001) = -60$dBc. Certainly the amplitude of the spur can be approximated this way.

While interesting there are difficulties in using this method to develop a spur cancellation routine. There is no real way of adjusting the phase of the spur relative to the desired signal. If there is some sort of phase component to the distortion it can not be determined in this manner. Multiplying by $1 \pm \epsilon$ only accounts for the generation of one of the spurs. The other spur will have to be modelled by some other method. A similar method to this has been used to improve the resolution of an RFDAC [16] but amplitude resolution is not a fundamental problem in this application.



(a) Undistorted CW



(b) CW Distorted by $(1 \pm \epsilon)$

Figure 2.3: Normal and Distorted CW's at $F_c = 850MHz$

2.2.4 Spurious Free Dynamic Range

Spurious Free Dynamic Range (SFDR) is defined as the ratio of the power in a sinusoid to the power in the largest spur measured with a spectrum analyzer at the output of the DAC. The sinusoid applied to the DAC is digitally generated. The SFDR is a function of the amplitude and frequency of the sinusoid and the frequency of the clock that drives the DAC. The data sheets usually have a few SFDR vs frequency graphs, each for a different clock rate. An example of a typical plot is shown in Figure 2.4. Each graph usually has

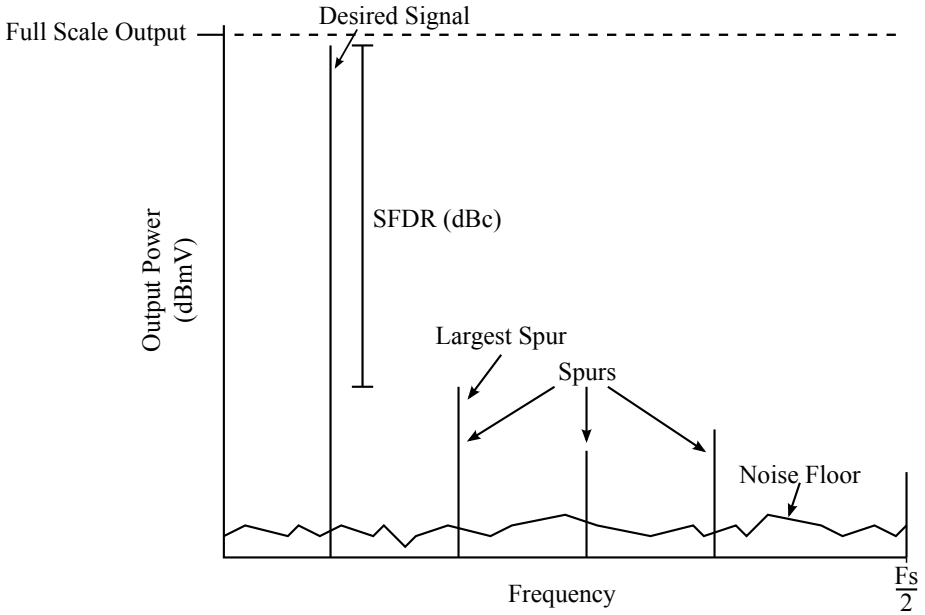


Figure 2.4: A Typical SFDR Plot

several curves showing the SFDR vs frequency for sinusoids with different amplitudes. The largest spur is usually, but not always, a harmonic of the sinusoid applied to the DAC.

2.2.5 Harmonic Distortion

The harmonic distortion is specified by three values: the total harmonic distortion (THD), second harmonic distortion ($HD2^{nd}$) and third harmonic distortion ($HD3^{rd}$). The THD is the largest possible ratio of the power in the output sinusoid to the sum of the powers in all the harmonics. The $HD2^{nd}$ is the largest possible ratio of the power in the stimulating sinusoid to the power in the second harmonic. The $HD3^{rd}$ is the largest possible ratio of the

power in the stimulating sinusoid to the power in the third harmonic. These specifications have similar information to SFDR, but SFDR is usually held in higher regard.

2.2.6 Intermodulation Distortion

When two sinusoids of frequency f_1 and f_2 are summed and applied to the DAC, non-linearities and glitches cause spurs at frequencies $2f_1$, $2f_2$, $f_1 \pm f_2$, $3f_1$, $3f_2$, $2f_1 \pm f_2$ and $f_1 \pm 2f_2$. The spurs at frequencies $f_1 \pm f_2$, $2f_1 \pm f_2$ and $f_1 \pm 2f_2$ are called the intermodulation distortion (IMD). The power in these spurs is a function of frequency and amplitude of the stimulating sinusoids. The IMD is specified as the ratio of power in the two sinusoids to the maximum possible power in IMD.

IMD is a critical specification when the DAC is used in communications equipment that has either or both spectral mask and spurious emissions as specifications. Pre-distortion to compensate for intermodulation distortion is similar to what is used to improve the linearity of power amplifiers. Pre-distortion of this sort has been well documented but there is still research being done to find algorithms that determine the pre-distortion coefficients [17] [18].

2.3 Milestone Architectures used in Commercial DACs

The architecture of DACs has changed over time to suit ever changing and expanding applications and take advantage of ever changing technology. Today there are a wide variety of applications that use DACs, but no one DAC fits all applications. For example DACs used in instrumentation need not be fast, but must be very high resolution, whereas DACs used in communication must be fast, but need not be high resolution.

RFDACs are a recent development and it seems manufacturers have not yet released the architectures used in their products. The reasons for this are unknown, but are most likely related to guarding intellectual property that provides a competitive advantage. This section covers the milestone architectures of DACs that evolved into the modern day DACs used in the communications industry.

2.3.1 String DACs

String DACs date back to the mid 1800's with Lord Kelvin's voltage divider, but were first implemented circa 1920. Figure 2.5 shows the architecture for a string DAC. It is

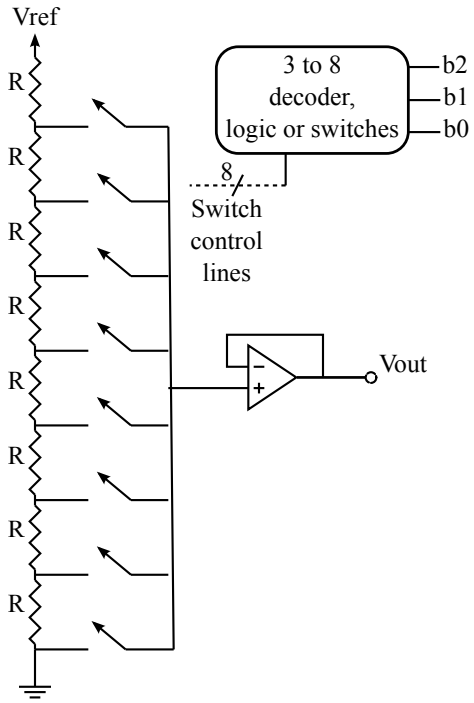


Figure 2.5: Architecture of a string DAC

simply a voltage divider with 2^N resistors, where N is the number of bits in the DAC. This architecture requires 2^N switches that were originally implemented with relays, but today they are usually implemented with bipolar or CMOS technology [19]. Only one switch is closed at a time to select a voltage from the voltage divider. Obviously a combinational logic circuit is required to decode the N-bit input into a one-hot 2^N-bit output.

The advantage of the string DAC is its DNL is very small. The DNL, when measured in LSBs, is basically the tolerance of the resistors. Such architectures yield relatively large SFDRs. The disadvantage is that 2^N resistors and switches are required, which becomes excessive for N greater than 10 or so.

String DACs are sometimes called thermometer DACs or voltage thermometer DACs. This name is not particularly fitting, but one could imagine the switch that is closed repre-

senting the mercury level in a thermometer.

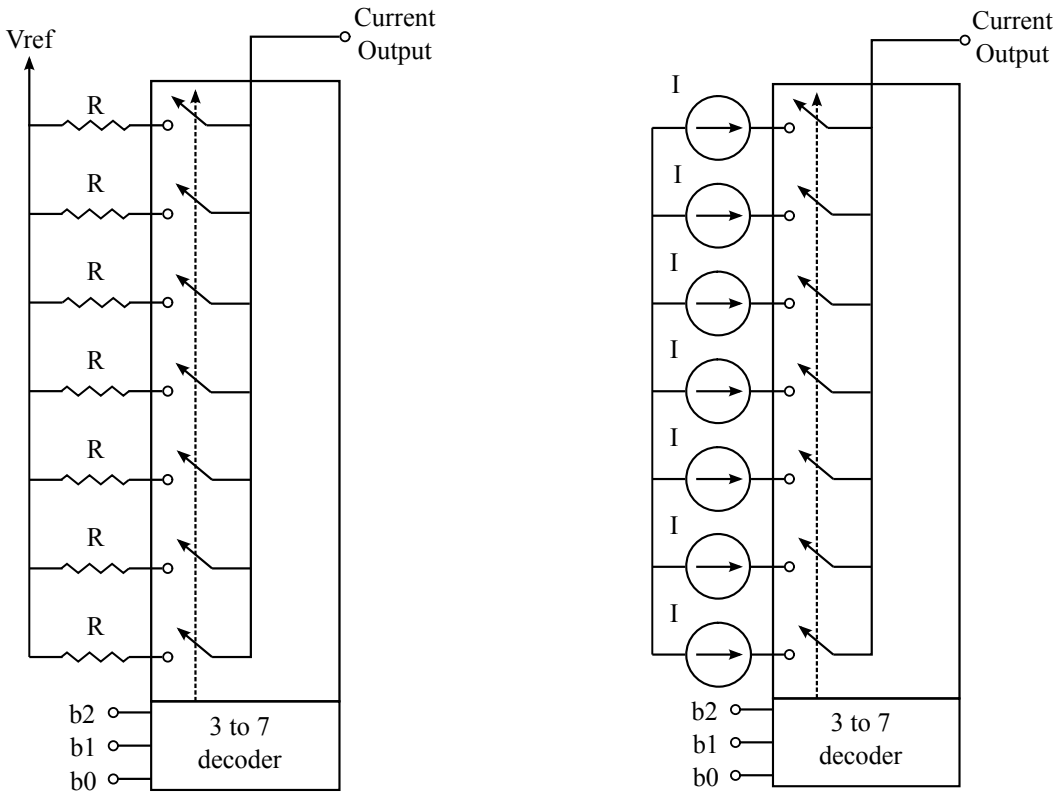
2.3.2 Current Output Thermometer DACs

A current-output DAC that is the counter part to the voltage-output string DAC is called a current output thermometer DAC. The output of a current DAC is intended to be connected to virtual ground. Often it is connected to a current-to-voltage converter, which is the negative input of a operational amplifier that has its positive input connected to ground and a feedback resistor connected to the negative terminal. If using an op-amp is not practical due to the large output bandwidth the current output is often connected to a differential reconstruction filter that has a resistive termination to convert the current into power.

Current Output thermometer DACs can be built with either $2^N - 1$ resistors or $2^N - 1$ current sources as shown in Figure 2.6. In the big picture the input to the DAC is viewed as an unsigned fraction, but for purposes of explaining the operation of the thermometer DAC it will be viewed as an unsigned integer that can take on decimal equivalent values of 0, 1, ..., $2^N - 1$. The function of the DAC is to produce an output current proportional to the value of the digital input. If the input to the DAC has a decimal equivalent of 0, then no switches are closed. If the input is full scale, this decimal equivalent $2^N - 1$, then all the switches must be closed. If the input has decimal equivalent k, then k of the switches must be closed. It does not matter which switches are closed as long as the number of closed switches is exactly k. If the bottom k switches are closed, then the DAC resembles a thermometer with the closed switches representing the mercury.

This type of DAC has the same advantage and disadvantage as the string DAC. Its DNL in LSBs is the tolerance of the resistors or current sources and it requires basically the same number of switches and resistors or current sources.

The thermometer DAC does have one major advantage over the string DAC. If a cleaver algorithm is used for deciding what k switches to close the INL becomes randomized. The algorithm has to close a different combination of switches on subsequent occurrences of the number k. Doing so makes the input-output characteristic non-deterministic, which means



(a) Voltage Based

(b) Current Source Based

Figure 2.6: Architecture of a Thermometer DAC

it can no longer be be modelled by a third order polynomial. The input-output characteristic must be modelled as deterministic with a random noise component. Such a model is described by

$$y(t)\big|_{nT_s \le t \le (n+1)T_s} = x[n] + \gamma_k[n], \tag{2.2}$$

where $\gamma_k[n]$ is random noise drawn from a distribution that depends on k. Of course for $k = 0$ and $k = 2^N - 1$, $\gamma_k[n]$ is zero. The distribution of $\gamma_k[n]$ is approximately Gaussian for $15 < k < 2^N - 15$ and the variance is largest for $k = 2^N/2$.

Randomizing the selection of switches does not change the power in the error, but does change its spectral distribution. The distortion of a sinusoid will appear as random noise as opposed to a harmonic. Randomizing the selection of the switches significantly reduces IMD, HD2nd, HD3rd and SFDR.

The architectures shown in Figure 2.6 are not suited for high speed DACs for two reasons.

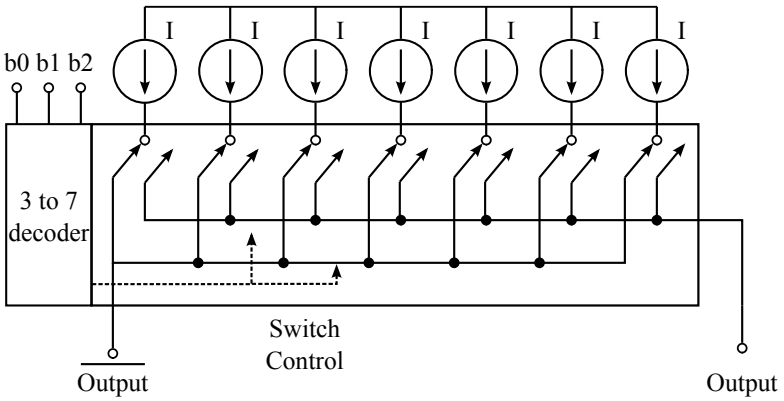


Figure 2.7: Architecture of a fast Thermometer DAC

Parasitic capacitance on the resistor side of the switch causes a glitch, that is the current to spikes, immediately after a switch is turned on. This parasitic capacitance also makes the settling time quite long. To remedy these problems the architecture of Figure 2.7 is used. The current sources are never turned off. They are dumped into a complementary output. To ensure the voltage across the switches never changes the switches must be of make-before-break type. This eliminates the effect of the parasitic capacitance.

Another advantage of the architecture of Figure 2.7 is the complementary output can be used to reduce the effect of circuit board track inductance on the high frequency analog output and a differential output tends to reduce common mode noise on the output lines.

One design for an actual make or break type switch in an integrated circuit that has p-type transistors. Figure 2.8 shows a transistor circuit that switches a current source between

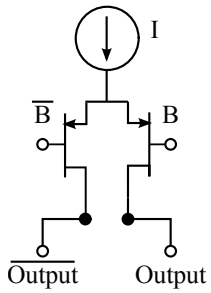


Figure 2.8: PMOS Transistor Current Switches

the output and it's complement. When the active bit is high the transistor on the right is turned on and current is routed to the output. When the bit is low the complement of that bit is high turning the left side switch on, routing current to the complement of the output. The make or break component in the switch rely's on the careful design of the timing and routing of both the active bit line and it's complement. A simple inverter will not suffice. An inverter will delay the complement of the bit meaning there will be a short time where neither or the switches are enabled. The signal routing must be done so a positive transition on either side of the switch will always proceed the negative transition of the complementary bit.

2.3.3 Binary Weighted Summing DACs

Binary weighting DACs are economic versions of thermometer DACs. The operating principle is the same, but the number of current sources is reduced from $2^N - 1$ to N. In principle his is done by bundling the current sources into groups of 1, 2, 4, ..., $2^N/2$ and switching each group in and out with a single switch driven by one of the bits in the N-bit input. In practice, to gain economic advantage, one current source replaces each bundle as shown in Figure 2.9.

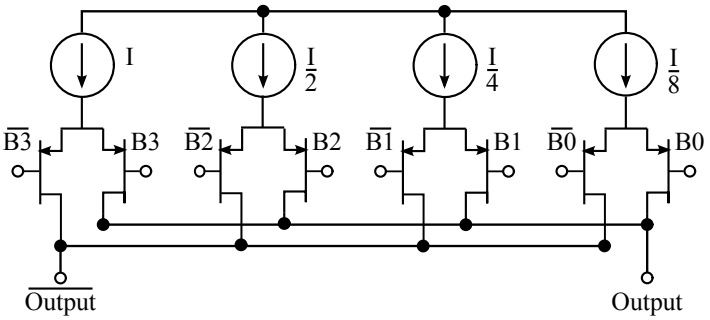


Figure 2.9: Architecture of a Binary Weighted Current Mode DAC

Unfortunately it is difficult to accurately scale the current sources in ICs at a ratio of even 128:1 [20]. The worst case error in each current source should be much less than the current that represents the least significant bit. If is very difficult to fabricate the large current sources with the necessary accuracy. Therefore, the DNL and INL are larger and of

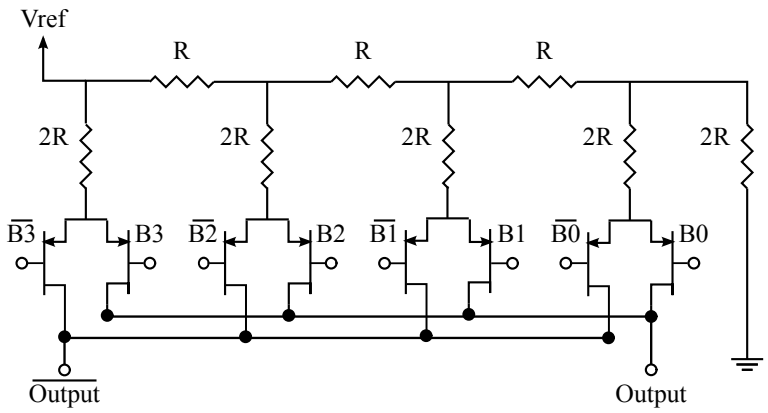


Figure 2.10: Architecture of a R-2R Ladder DAC

course SFDR is smaller than in thermometer DACS. For that reason this type of DAC is limited to about 10 bits.

2.3.4 Resistance Ladder (R-2R) DACs

A N-bit current mode DAC can be constructed from a network of $2N$ resistors as shown in Figure 2.10. The resistors have values R and 2R and are large enough in value to be individually laser trimmed in the factory. Again the switches are of make-before-break type so the voltage across the switches is only a few millivolts. This allows the reference voltage to be much much larger than the supply voltage for the DAC. When the DAC is implemented in CMOS technology the reference voltage can be of either polarity. For these reasons this is the most popular architecture for multiplying DACs.

2.3.5 Segmented DACs

The overall complexity of a high resolution DAC can be reduced by using two low resolution DACs connected in series, or at least connected in a way that for all intents and purposes is in series. The first DAC handles the most significant bits and the second handles the least significant bits. The two DACs may be of the same type or of different types. When two low resolution DACs are combined in this way to make a high resolution DAC the resulting DAC is called a segmented DAC.

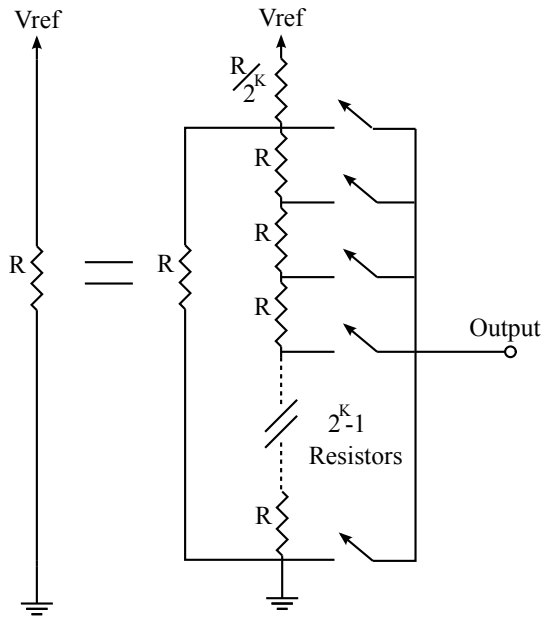


Figure 2.11: K-bit string DAC with end-to-end resistance R

Voltage Segmented DACs

There are a wide variety of segmented DACS. The one discussed here was filed for patent in 1997 by Dennis Dempsey and Christopher Gorman [21]. The design is a string DAC segmented with another string DAC. The intent is to make a N-bit string DAC from a M-bit string DAC and a K-bit string DAC, where $N = M + K$. The economic gain is in the reduction of the number of resistors from 2^N to $2^M + 2^K$. For $K = M = N/2$ the number of resistors required is reduced by a factor of $2^{N/2}/2$, which is large. Put another way, a segmented string DAC with $2N - 2$ bits of resolution can be constructed with the number resistors in an N-bit string DAC.

The underlying concept is to make an M-bit string DAC with switches at each voltage tap point so that the K-bit string DAC can be connected in parallel with any one of the M resistors. The resistor where the K-bit DAC is connected across is selected by the M most significant bits of the N-bit input and the tap in the K bit DAC that is to be used as the output is selected by the least significant K bits of the input. The ingenuity in the patent was in the design of K-bit DAC illustrated in Figure 2.11 as well as the idea to use switches

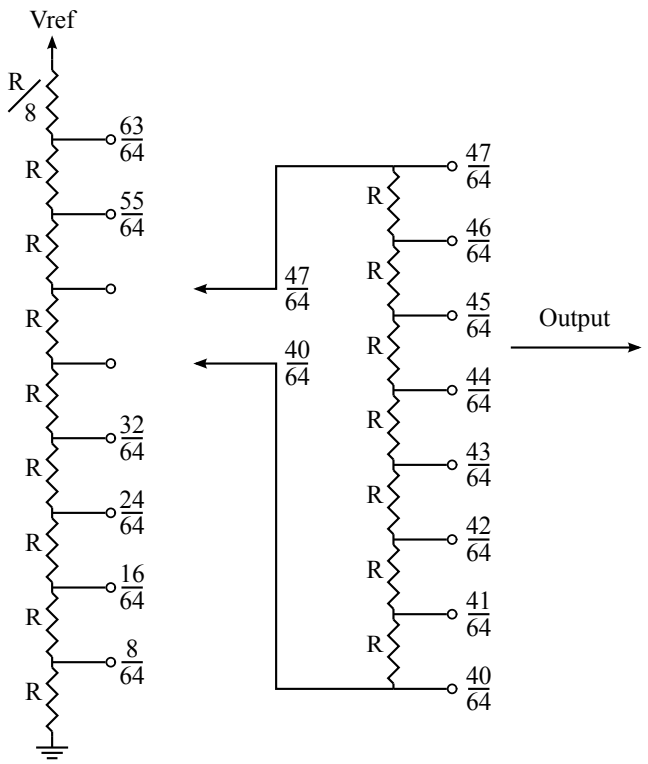


Figure 2.12: 6-bit Segmented string DAC using two 3-bit string DACS

so the K-bit DAC can be used as the LSB's for any possible MSB value.

The K-bit DAC has a series string of $2^K - 1$ resistors, all with resistance R, connected in parallel with a resistor with resistance R and all of that connected in series with a resistor with resistance $R/2^K$. The end-to-end resistance of the K-bit DAC is given by

$$\begin{aligned} R_{K\_DAC} &= \frac{R}{2^k} + \frac{R \times (2^K - 1)R}{R + (2^K - 1)R} \\ &= R. \end{aligned} \tag{2.3}$$

Such a construction allows the $R/2^K$ resistor to be removed from the K-bit DAC and moved to the top of the string of the M-bit DAC. Moving the $R/2^K$ resistor to the M-bit DAC allows the segmented DAC to be constructed as shown in Figure 2.12. The resistor in the M-bit DAC becomes the parallel R the K-bit DAC requires.

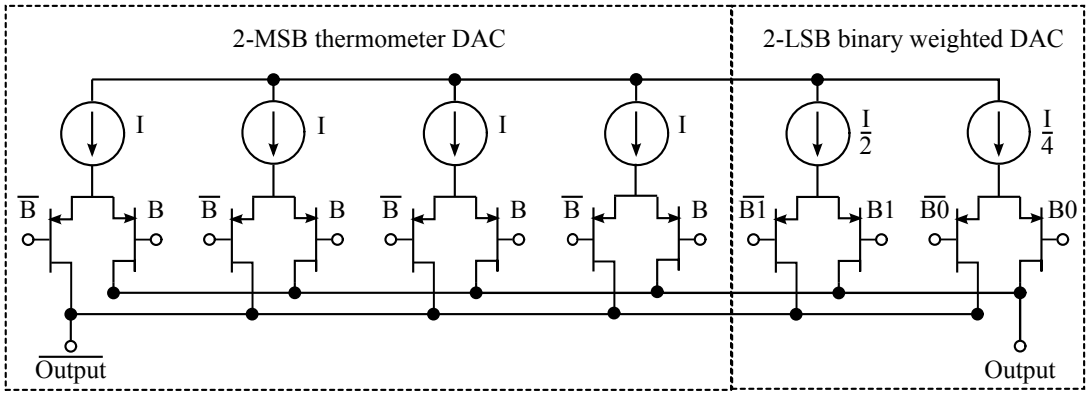


Figure 2.13: 4-bit Segmented Current Output DAC

Current Output Segmented DACs

Most if not all of the high-speed high-resolution DACs are segmented DACs and most of these that find an application as a transmitter in the communications industry are current output segmented DACs. The segmentation is not necessarily done with the same type of DAC. An example of a 4 bit segmented DAC constructed using a 2-bit Binary weighted DAC for the 2 LSB's and a 2-bit thermometer DAC for the 2 MSB's is illustrated in Figure 2.13. These combinations can also be further segmented by repeating the same DAC structure for groups of bits. The DAC in Figure 2.13 could be used to create an 8 bit DAC by repeating the same structure twice and attenuating the current output of the LSB's by 16:1. All of this is done to reduce the effect of resistor and current source tolerances while maintaining monotonicity and minimizing distortion and spur inducing glitches.

2.4 Radio Frequency DACs

There are several radio frequency DAC's currently available in the market. A short list includes: the AD9737 and AD9739 from Analog Devices, with 12 and 14 bits of resolution respectively and a maximum clock frequency of 2.5GHz, the DAC5670 from Texas Instruments with 14 bits of resolution and a maximum clock frequency of 2.4GHz, and the MAX5881 (12bits, 4.3GHz) and MAX5879 (14bits 2.1GHz) from Maxim Integrated Products Inc. These DAC's also often have the capability to function in a return-to-zero mode

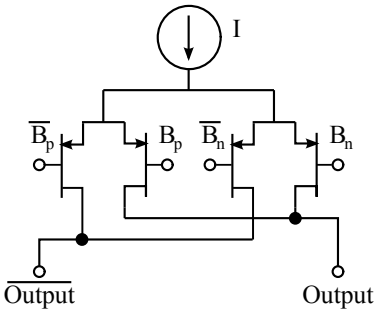


Figure 2.14: Quad-Switch Structure

that changes the shape of the output spectrum [22] so the $\frac{sin(x)}{x}$ roll-off is the same as a DAC with twice the sampling rate.

Due to the highly competitive nature of the semi-conductor industry there is very little information currently available detailing the internal structure of RFDAC's. The data-sheets indicate that they are current output DAC's and it can be surmised that they are segmented DAC's but the nature of each segment is unknown. One clue to the internal structure of the Analog Devices RFDAC's is offered in the data-sheet for the AD9737 and AD9739 where reference is made to a "Quad-Switch" structure. The quad switch uses a slightly more complicated structure than shown in Figure 2.8. "The quad-switch architecture masks the code-dependent glitches that occur in a conventional two-switch DAC." [23] This structure, shown in Figure 2.14 routes the current through different switches for each half of the DAC sampling clock cycle. If the bit is a 1 the current will be routed through B_p for the positive half of the sampling clock duty cycle and routed through B_n for the negative half. There are other examples of a "Quad-Switch" architecture found in available literature [24] [25] therefore it is likely that this switching style is often found in RFDAC's.

The indication is that the causes of glitches and spurs detailed in this section are similar to the causes of glitches in RFDAC's and the designers are working to minimize the distortion as much as possible. Without detailed knowledge of the internal structure of the RFDAC's a solution must be found that doesn't concentrate on one specific cause of distortion. Therefore the DAC being studied should be treated as a black-box. Doing this should produce a solution that will work with any DAC regardless of structure or manufacturer.

3. RFDAC Distortion Model and Parameterization

3.1 Modelling the RF DAC as a sum of desired and unwanted signals

The first stage of the problem is to model an RFDAC without detailed knowledge of its internal circuitry. The DAC will be modelled as the summation of the output of an ideal DAC (one that is distortion free) and distortion products that must be parameterized in some way. The unwanted distortion products are clearly visible if a sinusoid is used as a test signal.

Figure 3.1 shows the output spectrum of an RFDAC when a discrete sinusoidal test signal defined as $x[n] = A\cos(2\pi f_c n + \phi)$ is used. Once this discrete test signal is converted by the DAC to an analog signal it is defined as $x(t)$. The frequency of the sinusoid is near the upper edge of the band for CATV applications and the sampling rate is at the upper range allowed as an input to the RFDAC.

The span of the spectrum analyzer is set to start at 50MHz and stop at 1050MHz. Marker 1 indicates the resultant $x(t)$ with an analog frequency of $F_c = 850$MHz from using the test signal x[n] with a discrete frequency f_c of $0.3405\,{}^{cycles}/_{sample}$ and sampling frequency $F_s = 2496$MHz . The two main distortion products are identified by markers 2R and 3R. The marker table at the bottom of Figure 3.1 shows the amplitude and frequency of the signals identified by the markers. The 2Δ and 3Δ indicators are not relevant.

The distortion at marker 2 is related to the 2^{nd} harmonic of $x(t)$. Signals related to the second harmonic of $x(t)$ can now be defined as $x_{2H}(t)$. The frequency of the 2^{nd} harmonic

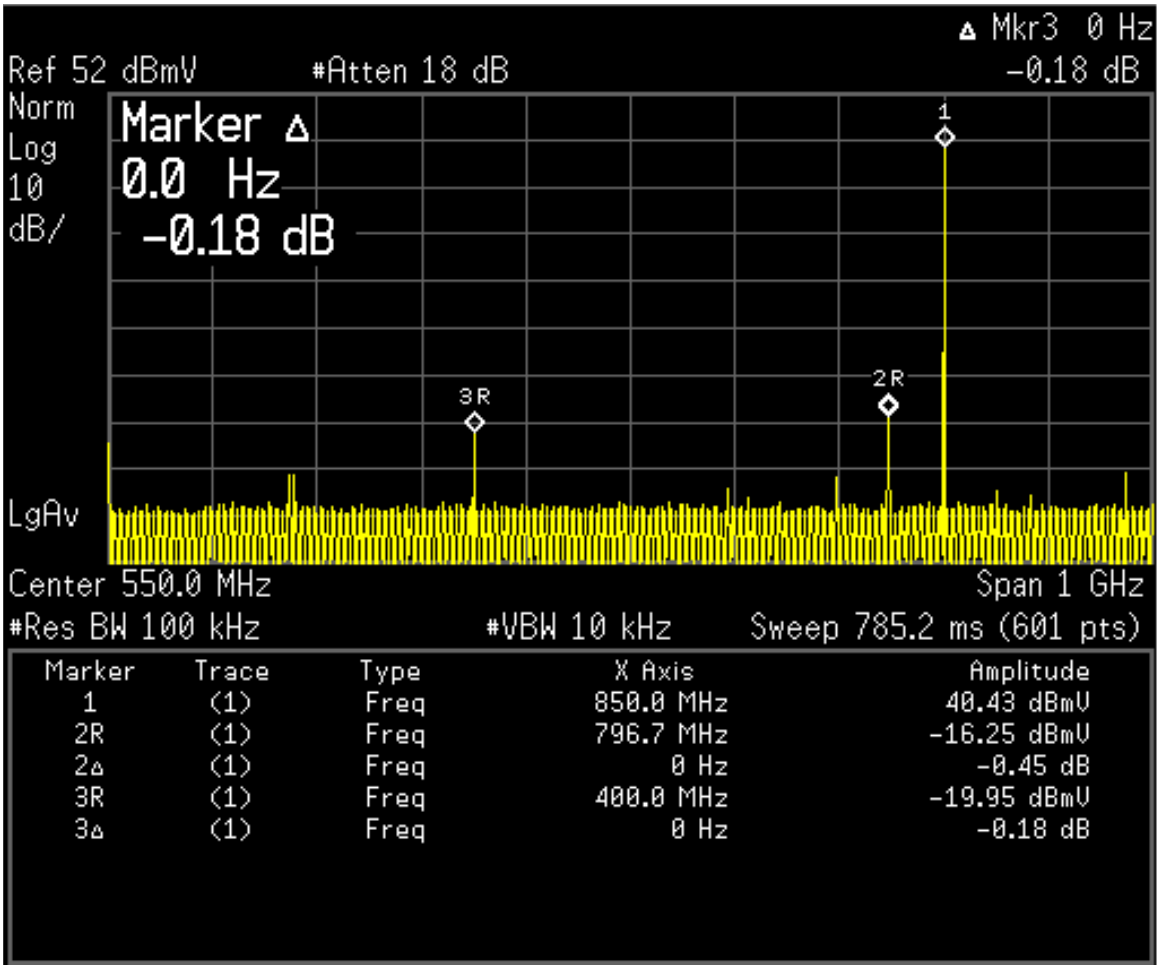


Figure 3.1: Continuous Wave output of an RFDAC.

of $x(t)$ is at $2F_c$ or 1700MHz. In a normal analog system the 2nd harmonic energy $x_{2H}(t)$ would only appear at 1700MHz. However, since this distortion is created inside the DAC and beyond the first Nyquist zone, given by $F_s/2$ the energy is indistinguishable from energy inside the first Nyquist zone and appears at $F_s - 2F_c = 2496\text{MHz} - 2(850\text{MHz})$ or 796MHz.

The analog sinusoid frequency, F_c, determines whether the second harmonic shows up in-band, i.e. in the CATV band governed by the DRFI specification, out of band or if it's alias shows up in band. The 2nd harmonic content will show up in band if F_c is less than half the CATV band edge of 1002MHz/2 or 501MHz. If the frequency of F_c is greater than 501MHz and less than 747MHz the 2nd harmonic content will be out of the CATV band, i.e. greater than 1002MHz, where the specification is relaxed. Once F_c surpasses 747MHz the

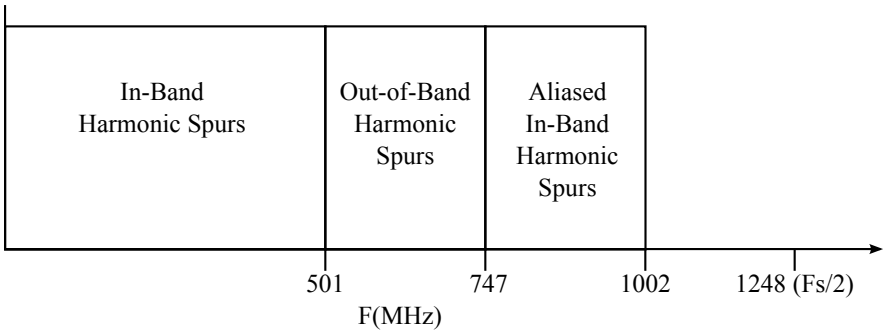


Figure 3.2: Zones where the $x_{2H}(t)$ spur falls in-band

alias will show up in band since $2496\text{MHz} - 2(747\text{MHz}) = 1002\text{MHz}$. These zones are shown in a frequency plot in Figure 3.2.

The distortion at marker 3 is related to half the sampling frequency and is likely due to the RFDAC using dual edges of the sampling clock where new samples are clocked into the DAC at a rate of $F_s/2$. This distortion appears at a frequency given by $F_s/2 - F_c = 2496/2\text{MHz} - 850\text{MHz}$ or 398MHz. Distortion that appears at this frequency will be defined as $x_{F_s/2-F_c}(t)$. This spur appears in band when F_c is between $(1248\text{MHz} - 1002\text{MHz}) = 246\text{MHz}$ to 1002MHz.

3.1.1 Cancellation Signal Frequencies

Distortion signal $x_{2H}(t)$ appears at the minimum of $2F_c$ and the alias $F_s - 2F_c$. Distortion signal $x_{F_s/2-F_c}(t)$ appears at frequency of $F_s/2 - F_c$. Since the distortion signal frequencies are always known it should be possible to create a discrete cancellation signal at those frequencies to remove the distortion. To remove the distortion a cancellation signal must have the same amplitude as the distortion and be exactly π radians out of phase. The cancellation signals are defined as $x_{2H}[n]$ and $x_{F_s/2-F_c}[n]$. In the case of $x_{2H}[n]$ the distortion can be generated by squaring $x[n]$ as shown below.

$$\begin{aligned} x_{2H}[n] &= \cos^2(2\pi F_c n/F_s + \phi) \\ &= 1/2(1 + \cos(2(2\pi F_c n/F_s + \phi))) \end{aligned} \tag{3.1}$$

The DC component can be subtracted after the squaring operation is performed.

To generate a signal at the correct frequency to cancel $x_{F_s/2-F_c}(t)$, the fact that $F_s/2 = \pi$ on a normalized discrete spectrum frequency plot is considered. Multiplying a discrete sinusoid $x[n]$ by $\cos(\pi n)$ should result in a cancellation signal at $F_s/2 - F_c$ as shown below.

$$\begin{aligned} x_{F_s/2-F_c}[n] &= \cos(2\pi F_c n/F_s + \phi)\cos(\pi n) \\ &= 1/2(\cos(\pi n + 2\pi F_c n/F_s + \phi) + \cos(\pi n - 2\pi F_c n/F_s - \phi)) \\ &= \cos((\pi - 2\pi F_c/F_s)n + \phi) \end{aligned} \tag{3.2}$$

Using these two mathematical operations to generate signals that cancel the distortion implies that the generation of the distortion inside the RFDAC can be modelled using the operations.

3.1.2 Cancellation Signal Amplitude and Phase

Figure 3.1 indicates that the amplitude of either of the distortion products or spurs is not equal to the test signal. It is empirically observed that the spur's amplitude changes as a function of the frequency of $x(t)$ and therefore $x[n]$. While there is no available information regarding the phase difference between each of the spurs and $x(t)$ it is assumed that a difference exists and is also a function of frequency. For this reason the actual analog output of the RFDAC can be modelled as a summation between the ideal distortion free output and distortion signals whose amplitude and phase are changed by a transfer function $H(e^{j\omega})$ then multiplied by the factors given in equations (3.1) and (3.2). A block diagram of of this model is given in Figure 3.3.

To design an effective cancellation network the frequency responses of the 2 distortion products must be measured as accurately as possible.

3.2 Test and Measurement Tools

3.2.1 Test System

The RFDAC chosen for the investigation is the AD9739 manufactured by Analog Devices. This particular RFDAC has 14 bits of resolution and a maximum sampling frequency of

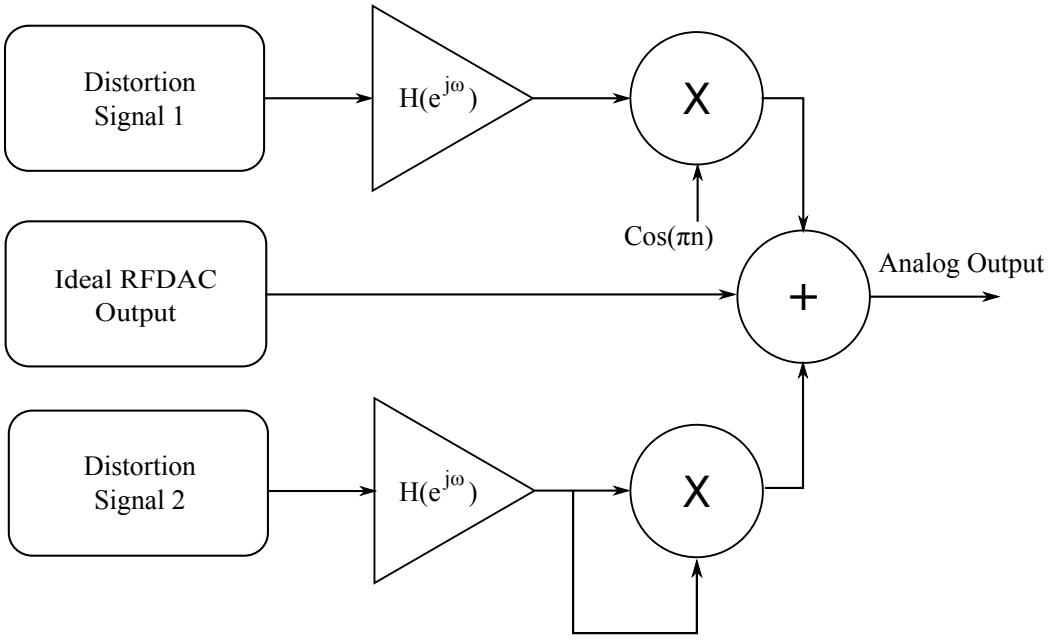


Figure 3.3: RFDAC Distortion Model

2.5GHz. In order to observe the output of the RFDAC the device itself is mounted on an evaluation board that has peripheral circuitry including connectors for inputs and outputs and components that regulate the power supply.

RFDAC's such as the AD9739 have a multitude of settings and configuration options available to a user. These settings are generally controlled through a serial peripheral interface (SPI) that allows configuration registers inside the RFDAC to be written to and read from. This interface is controlled through a USB connection with software that runs on a PC. A graphic user interface (GUI) allowing control of these registers is automatically downloaded from Analog Devices when the the RFDAC evaluation board is first connected.

The DAC evaluation board requires a few external inputs, including a datasource, and clock to facilitate operation. Naturally test equipment is required to observe the DAC output.

The datasource used to send a sample stream to the RFDAC is a piece of test equipment called the Data Pattern Generator or (DPG) also manufactured by Analog Devices. The DPG interfaces to the DAC evaluation board through a high speed parallel digital data connector. The DPG is a necessary component for measuring the RFDAC performance

since a signal of some sort must be provided to the evaluation board if any performance parameters are to be measured.

The DPG is controlled through a USB port using a downloaded GUI . Using the GUI, a user can generate a discrete sinusoid at a desired frequency, or 1 of several available standard signals including 6MHz bandwidth QAM signals that conform to the DOCSIS standard. Several CW's or QAM signals at multiple frequencies can also be combined together. The DPG has a feature that allows samples in the form of a text file to be loaded and sent to the DAC in a loop to generate a continuous periodic signal.

The sampling clock input of the AD9739 must be of very high quality. The clock is connected to the RFDAC board using a 50ohm coaxial cable. The external clock is a frequency synthesizer with an integrated VCO that has a low noise Temperature Compensated Crystal Oscillator (TCXO) reference. The reference in this case has a frequency of 48MHz. The sampling frequency of 2496MHz was selected because because it is near the maximum for the AD9739 and is an integer multiple of this reference. Such modules are commercially available and can also be obtained on an evaluation board.

Using a sampling clock at this frequency to generate DRFI compliant signals means the lowest frequency image appears at $F_s - 1002$MHz or 1494MHz. The DRFI spec for spurious emissions that are at a frequency higher than 1002MHz, but below 3000MHz, is -55dB with respect to an active in-band channel. The output of the RFDAC is simply terminated with a 50ohm resistor without a reconstruction filter to remove the unwanted images.

Power is supplied to the AD9739 and frequency synthesizer boards with a standard bench DC supply. There are voltage regulators on each of the test boards ensuring both a low noise DC source and that the correct voltages are applied to the various power supply pins on the parts. The DPG is powered by the 60Hz 110V wall supply. Figure 3.4 details a block diagram of the test set-up, while Figure 3.5 shows a photograph of the test equipment.

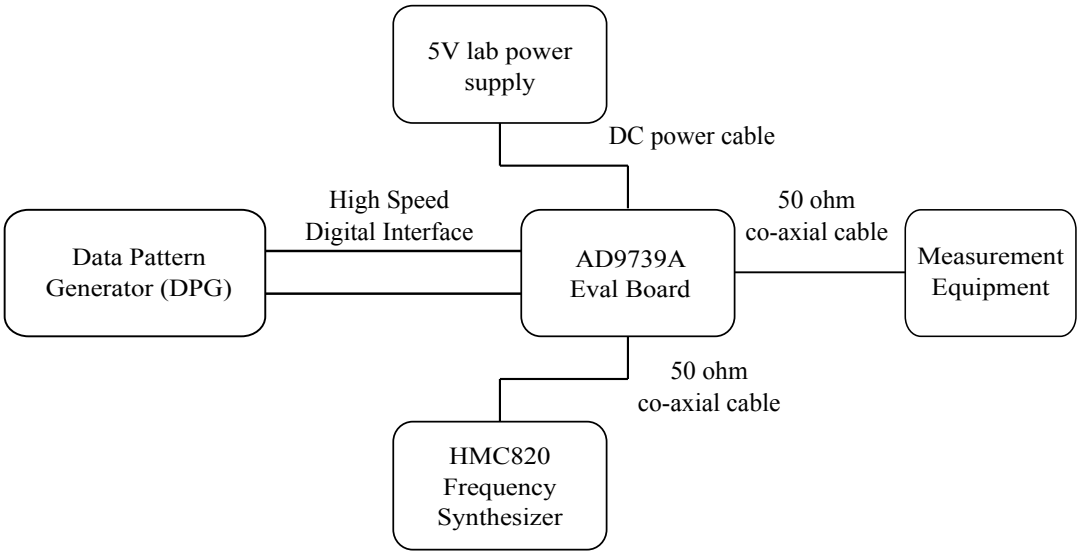


Figure 3.4: Block Diagram of Test and Measurement equipment.

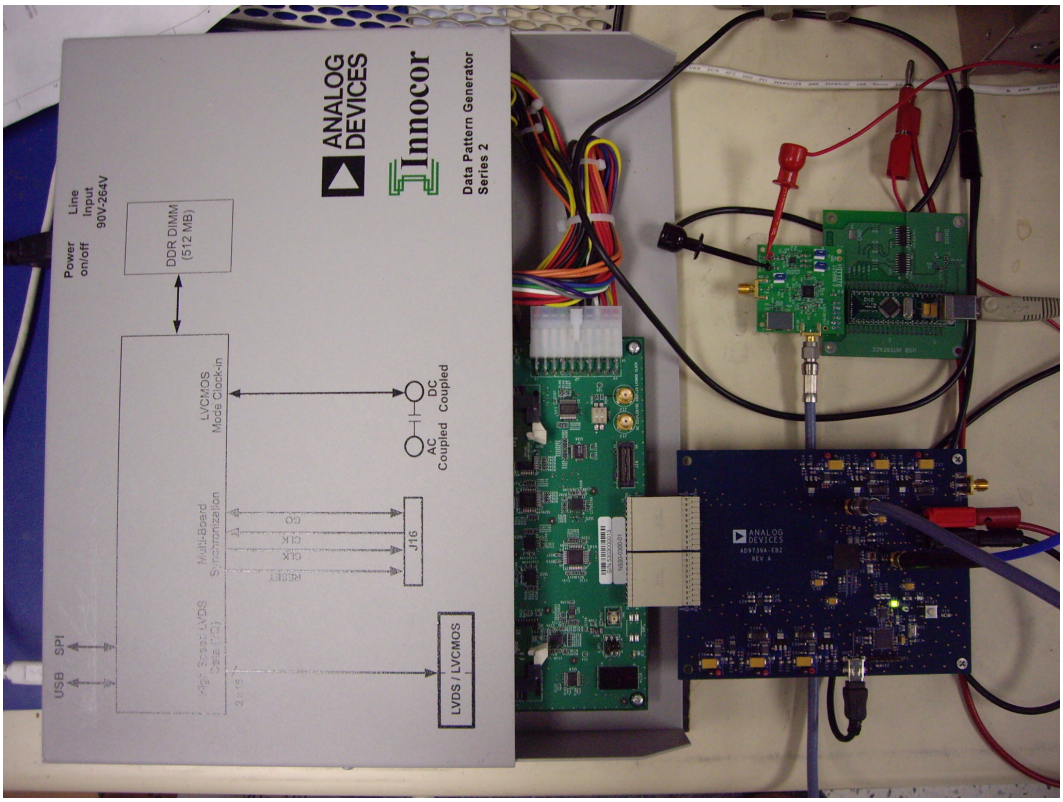


Figure 3.5: Test Equipment.

3.3 Measurement Tools and Methods

A method must be developed where the distortion products can be measured relative to the desired signal. The distortion amplitude can be measured in absolute terms or relative to the expected output signal. The phase measurement must be relative to the expected output signal. These measurements can be made in either the time domain or the frequency domain.

3.3.1 Time Domain Measurement

Oscilloscope

To make a time domain measurement of a signal the most common tool to use is an oscilloscope. There are two fundamental problems when measurements of this nature are being made, the bandwidth and the dynamic range of the measurement instrument. The necessary bandwidth of 1002MHz is very wide for most oscilloscopes, but there are high performance models that can achieve it. An Infinium 54846A oscilloscope manufactured by Agilent Technologies has an effective bandwidth of 2.25GHz and could be suitable.

These oscilloscopes have a feature where the data samples can be captured and stored for analysis. A discrete time Fourier transform can then be used to convert the samples to the frequency domain where the complex spectral components of the distortion signals could be analyzed in relation to the expected signal.

After making these measurements it was found that the dynamic range of the oscilloscope is not sufficient to measure the distortion products along with the test CW. When the discrete time Fourier transform was performed the distortion signals were in the noise floor. For this reason the oscilloscope was abandoned as a measurement tool.

Vector Signal Analyzer

The second tool considered for making a time domain measurement is a Vector Signal Analyzer (VSA). A VSA is generally used to demodulate a signal, but the option is available to make time domain sample captures. A VSA usually needs a high performance receiver,

most commonly a spectrum analyzer as an input. This input is commonly referred to as the RF head-end. The VSA then down-converts a fixed bandwidth to baseband where the de-modulated signal can be analyzed. The inherent problem with VSA's are their bandwidth.

After attempting to make time domain measurements with a VSA it was determined that the capture range of time domain signals that can be exported for analysis is too small to be very useful. The limited bandwidth means that any useful measurements would have to be made when the the distortion spurs could be placed in the same 36MHz bandwidth as the desired signal. For these reasons the VSA was also abandoned as a measurement tool.

3.3.2 Frequency Domain Measurement

Of all the equipment available for frequency domain measurements the instrument most commonly used is a spectrum analyzer. The screen capture taken in Figure 3.1 was taken with a PSA series E4440A spectrum analyzer from Agilent Technologies. The PSA is almost an industry standard for high quality spectrum analysis and is used throughout the CATV industry. In many cases the E4440A is the spectrum analyzer used by various test labs to certify that a product is compliant with the DOCSIS specification.

There is no fundamental problem with the measurement bandwidth using this spectrum analyzer. The internal operation makes a sweeping power measurement at the output of a narrowband filter and displays power vs. frequency [26]. The only limit to the bandwidth that can be displayed is the frequency range that the analyzer can operate over. Due to these considerations all measurements were made in this project with a spectrum analyzer.

3.3.3 Sample Generation

A method of generating the test signals at F_c must be developed. The test CW is generated by repeatedly sequencing through a buffer of length 2^{14} stored values. The input is not a pure sinusoid as it must be quantized to the number of input bits used by the RFDAC, in this case 14 bits. The sinusoid prior to quantization has the form.

$$x[n] = A\cos(2\pi f_0 n + \phi)$$

where A, f_0 and ϕ should be chosen to minimize any spurs caused by quantization and hit all or nearly all the possible input levels of the DAC.

To reach the highest input level of the RFDAC, the amplitude must be $A = 2^{14} - 1$. To minimize the spurs caused by quantization noise, the phase $\theta[n] = 2\pi f_0 n + \phi$ should be unique for every sample in the buffer. $x[n]$ must be periodic with a period of 2^{14} since that is the length of the buffer. That being said f_0 must be a multiple of the fundamental frequency, which is $\frac{1}{2^{14}}$ $^{\text{cycles}}/_{\text{sample}}$. Therefore

$$\theta[n] = 2\pi\frac{M}{2^{14}}n + \phi$$

where M is an integer. If M is even,

$$\theta[n] = \theta[n + \frac{1}{2}2^{14}] + \text{a multiple of } 2\pi$$

The reason for this is

$$2\pi\frac{M}{2^{14}}(\frac{2^{14}}{2}) = 2\pi\frac{M}{2}$$

and $\frac{M}{2}$ is an integer. Therefore, if M is odd, the smallest value of K for which

$$\theta[n] = \theta[n + K] + \text{a multiple of } 2\pi$$

is $K = 2^{14}$. Choosing M to be odd produces 2^{14} phases.

The last parameter to consider is ϕ. While the selection of M being odd ensures 2^{14} different phases it does not ensure 2^{14} different values for $x[n]$. Take $x[n]$ and $x[m]$ where $m = 2^{14} - n$.

$$x[n] = A\cos(2\pi\frac{M}{2^{14}}n + \phi)$$

and

$$\begin{aligned} x[m] &= A\cos(2\pi\frac{M}{2^{14}}(2^{14} - n) + \phi) \\ &= A\cos(2\pi M - \frac{2\pi n}{2^{14}} + \phi) \\ &= A\cos(-\frac{2\pi n}{2^{14}} + \phi) \\ &= A\cos(\frac{2\pi n}{2^{14}} - \phi) \end{aligned} \tag{3.3}$$

If $\phi = 0$, then $x[n] = x[m]$. However if $\phi = \frac{1}{4}(\frac{2\pi}{2^{14}})$, which is $^1/_4$ the spacing of the set of 2^{14} phases, all $x[n]$ will be unique.

In summary the frequency may be $\frac{M}{2^{14}}$ for M odd. The amplitude and phase should be $A = 1 - 2^{-14}$ and $\phi = \frac{2\pi}{4 \cdot 2^{14}}$.

To design around these constraints the desired output frequency is used as a parameter in a formula that calculates the actual discrete output frequency. To ensure there is no discontinuity in the sampled waveform when the end of the file is reached and the DPG loops back to the beginning there must be an integer number of cycles per 16384 samples.

A Matlab script is then used to generate the 16384 samples that are uploaded and output in a continuous loop by the DPG. Running this test loop resulted in a clean signal. The final script used to generate test CW signals is given below.

```
% Function to create a discrete cos wave
% Used to create test tones for the DPG
% Fc = Nominal centre frequency
% Fs = Sampling frequency
% Scaler = amount CW is scaled from full-scale

function [Xs_nom] = CW_gen(Fc, Fs, Scaler)

nom_freq1 = max(primes(Fc*16384/Fs))/16384*Fs; %Actual frequency found
%Genreate cos wave loop of 16384 samples
for n = 0:16383;

    Xs_nom(n+1) = Scaler*cos((2*pi*n)*nom_freq1/Fs);
end
```

To illustrate the difference between the calculated frequency and an exact desired frequency the first 8 discrete samples of a cosine at 624MHz or $\pi/4$ are used. $[\ 1,\ 0,\ -1,\ 0,\ 1,\ 0,\ -1,\ 0\ ]$. In this case there is no truncation or rounding of the discrete samples so there will be no quantizing noise at all. This is compared to samples generated by the function. The calcu-

lated frequency is 623.5430MHz and the first 8 generated samples are

$$[\,1.0000,\ 0.0012,\ -1.0000,\ -0.0035,\ 1.0000,\ 0.0058,\ -1.0000,\ -0.0080].$$

Unlike the first series, the samples are not just the same values continuously repeated. In this case since both the original desired frequency and the calculated frequency will have an integer number of cycles and there will be no discontinuity between samples 16384 and 1.

3.4 Distortion Measurement

3.4.1 Measuring the $x_{F_s/2-F_c}[n]$ cancellation parameters

An initial assumption is made that the spurs can be cancelled independently from each other. To model the frequency response of the distortion, a process is developed to accurately measure its amplitude and phase offset relative to to $x[n]$. The measurements are repeated at regular frequency intervals across the band of interest resulting in data points that give the frequency response. The procedure used to make these measurements is detailed below.

1. Another script is used that adds a second discrete sinusoid at a different frequency and has a scaled amplitude relative to $x[n]$. The 2nd frequency used should have a slight offset of approximately 200KHz from the spurious signal so both signals can be seen on the spectrum analyzer.

2. The amplitude of the 2nd signal is adjusted until it is exactly the same as the spur. This measured amplitude will be used as a starting point for the amplitude of the cancellation signal.

3. A third script to apply the cancellation signal is now used. This time the cancellation signal is created using the multiplication shown in (3.2). A block diagram showing the cancellation structure is illustrated in Figure 3.6. The subscript c is used to show the discrete cancellation sinusoid.

4. An initial check is made to find how much the spur is reduced by simply subtracting the cancellation signal from the desired signal. This technique will verify that the

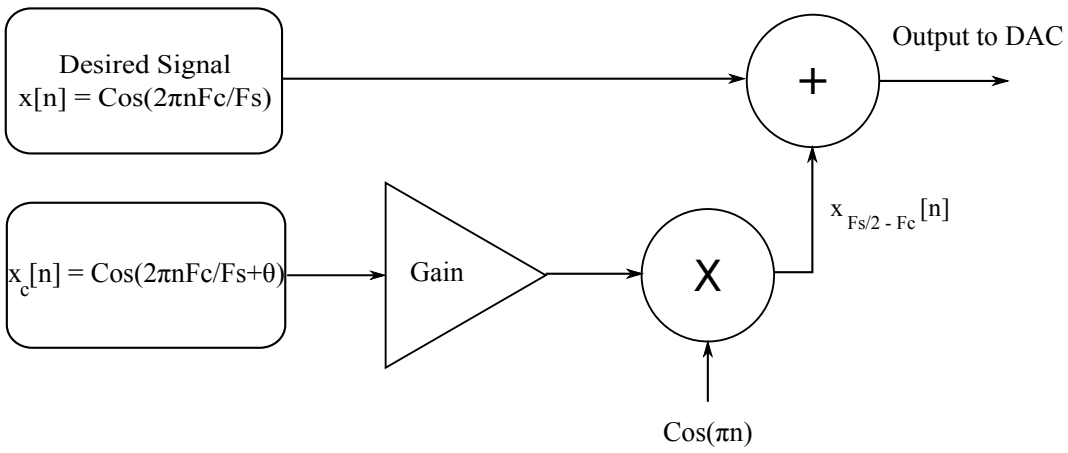


Figure 3.6: Block Diagram of $x_{F_s/2-F_c}(t)$ spur cancellation

cancellation signal is at the correct frequency and that there is some validity to the method.

5. The amplitude is now adjusted until the spur is reduced as much as possible. After the first amplitude adjustment is complete, the phase of the cancellation signal is adjusted until a new minimum amplitude for the spur is obtained.

6. The process is repeated alternately adjusting the amplitude and phase by trial and error to minimize the spur level to as low a level as is reasonable. A 30dB reduction is chosen as an initial goal.

7. After the measurement is made at an initial frequency the process is repeated, changing the frequency of $x[n]$ across the band to determine the frequency response of the distortion signal.

Results

The measured results for $x_{F_s/2-F_c}(t)$ are very good. The spur can be reduced by 30dB over the entire CATV band. Data points are taken with a frequency spacing of approximately 96MHz. A screen capture showing the effectiveness of the added cancellation signal in reducing the spur is shown in Figure 3.7. The spur amplitude is shown as the gain in dB used as a scaler multiplier on the distortion sinusoid. The results measured as a function of

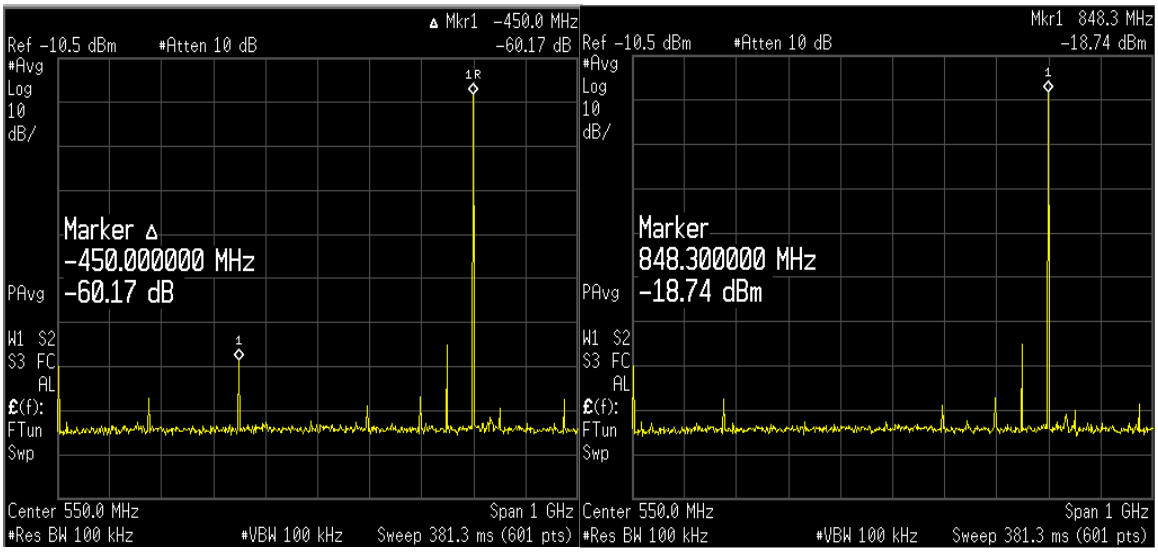


Figure 3.7: Screen Captures showing $x_{F_s/2-F_c}(t)$ spur cancellation

frequency are shown in Table 3.1.

Verification

After the frequency response of the cancellation signal has been measured, it is desired to gain confidence that the measurements made and data collected using a sinusoid will be useful when the desired signal is of the modulated DOCSIS type. Two different variations of the test signals are used.

First the amplitude of the required cancellation signal must change in direct proportion to the amplitude of the desired signal. If the amplitude change is linear the assumption can reasonably be made that the cancellation routine will work for all the various amplitude levels that will be present in a modulated signal. Table 3.2 shows the change in power of $x[n]$ relative to a full-scale signal and the relative amplitude level of the $x_{F_s/2-F_c}[n]$ distortion signal. The frequency F_c of $x[n]$ is always 798MHz.

The second verification step is to make sure the cancellation routine works when 2 discrete sinusoidal $x[n]$'s at a convenient frequency offset are combined and used as the test signal. A frequency separation of 6 MHz is selected to replicate the channel bandwidth of an Annex

Table 3.1: Amplitude and Phase measurements of $x_{F_s/2-F_c}[n]$

$x[n]$ Frequency (MHz)	$x_{F_s/2-F_c}[n]$ Gain (dB)	$x_{F_s/2-F_c}[n]$ θ (rad)
246	-64.4	0.15
390	-64.0	0.07
486	-63.8	0.038
579	-64.1	0.025
678	-63.8	0.012
773	-63.2	0.003
869	-62.6	0.00
965	-62.2	0.00
1002	-62.0	0.00

Table 3.2: $x_{F_s/2-F_c}(t)$ Spur Amplitude with and without Cancellation

$x[n]$ Amp. (dBFS)	$x_{F_s/2-F_c}(t)$ Amp. (dBc)	$x_{F_s/2-F_c}(t)$ Amp. (dBc) with Cancellation
0	-60.3	-88
-3	-60.5	-87
-6	-60.5	-85

B DOCSIS signal.

If the cancellation algorithm is successful in eliminating the distortion caused by two signals separated by a 6MHz channel bandwidth, it should also work for a 6MHz bandwidth modulated signal. The indication is that there is no dramatic change in the distortion signal frequency response over a narrow band. Using sinusoidal $x[n]$'s at 866MHz and 872MHz with pre-distortion the $x_{F_s/2-F_c}(t)$ spurs are attenuated into the noise floor.

3.4.2 Measuring the $x_{2H}[n]$ cancellation parameters

With the success in finding the frequency response of the first spur a very similar method is used to try and measure the frequency response of the harmonic $x_{2H}[n]$ distortion signal.

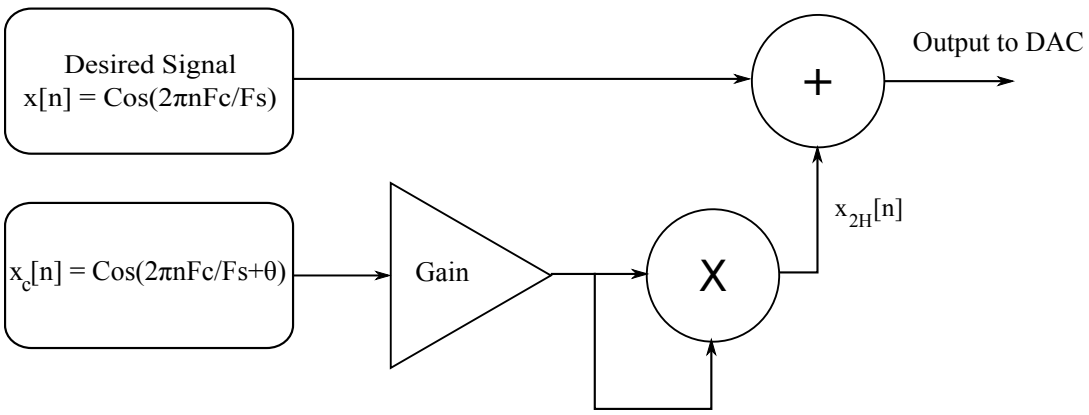


Figure 3.8: Block Diagram of $x_{2H}[n]$ spur cancellation

1. Using the same script as in point 1 of Subsection 3.4.1 the amplitude of the $x_{2H}[n]$ cancellation signal is estimated

2. The amplitude and phase adjusted version of the test signal is squared to generate the cancellation signal at the correct frequency as described in (3.1). Figure 3.8 shows a block diagram of the signals.

3. Points 5, 6, and 7 from subsection 3.4.1 are repeated with this spur.

Results

The results are also excellent for this spur. In general the harmonic related spur can be reduced by approximately 30dB. Compared to $x_{F_s/2-F_c}[n]$,a higher phase offset component is needed for $x_{2H}[n]$ to cancel out the $x_{2H}(t)$. Measured results as a function frequency are detailed in Table 3.3.

Verification

The cancellation of this spur is also verified to be useful for a modulated signal. Verification using 2 $x[n]$'s is performed as part of the measurements. The results showing the amplitude level of $x[n]$ dB relative to a full scale output vs. $x_{2H}(t)$ power, relative to $x(t)$ is detailed in Table 3.4. The measurements are made at 798MHz. Note that the level of the

Table 3.3: Amplitude and Phase measurements of $x_{2H}[n]$ spur

$x[n]$ Frequency (MHz)	$x_{2H}[n]$ Gain (dB)	$x_{2H}[n]$ θ (rad)
102	-34.5	0.42
150	-34.3	0.18
192	-34.5	0.42
246	-32.1	0.26
291	-30.4	0.20
342	-29.0	0.18
390	-27.7	0.14
440	-26.5	0.12
486	-25.8	0.10
749	-24.9	0.45
773	-25.3	0.45
798	-25.3	0.43
822	-25.4	0.43
869	-25.5	0.45
916	-25.6	0.40
965	-25.3	0.40
1002	-26.2	0.34

spur is decreased by 6dB for every 3dB reduction of the desired signal. The extra 3dB drop occurs when the reduced amplitude test signal is squared to generate the harmonic content.

Table 3.4: $x_{2H}(t)$ Spur Amplitude with and without Cancellation

$x[n]$ Amp. (dBFS)	$x_{2H}(t)$ Amp. (dBc)	$x_{2H}(t)$ Amp. with Cancellation
0	-56.8	-79
-3	-60.1	-82
-6	-63.4	-85
-9	-65.9	-85

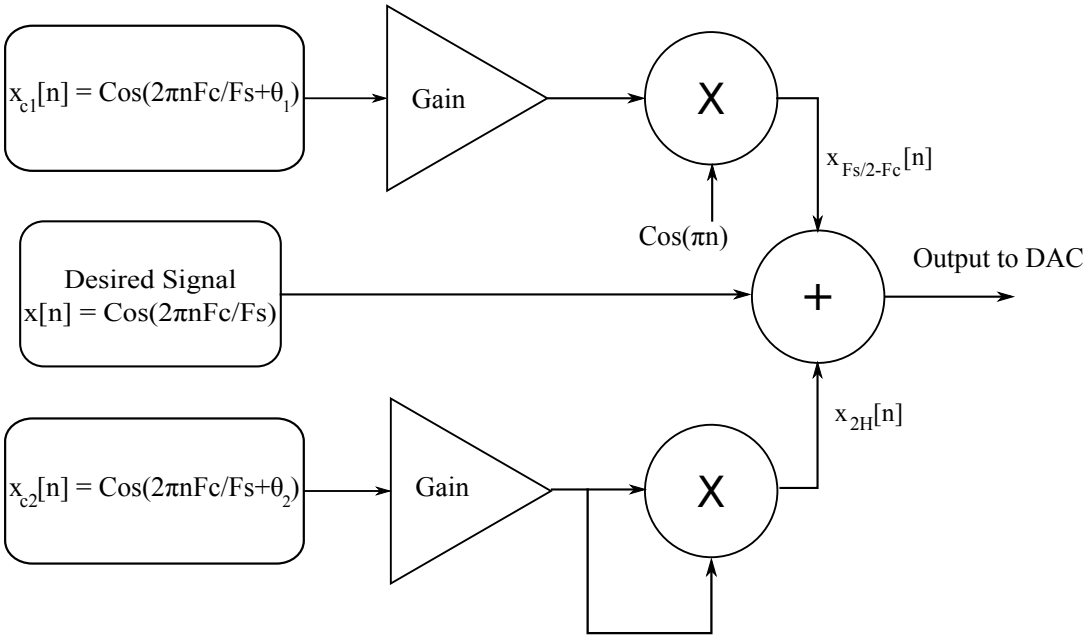


Figure 3.9: Block Diagram of final cancellation algorithm.

3.4.3 Combining the two Cancellation signals

The distortion model validity is confirmed when the generation of both cancellation signals is combined into one script and the measurements repeated. Figure 3.9 details a block diagram of the final algorithm used to cancel both of the distortion signals enabling the frequency responses to be measured.

An example of the results at selected output frequencies of 869MHz and 486MHz are shown in Figure 3.10 and Figure 3.11. The results across frequency are compared to those in Tables 3.1 and 3.3 and found to be identical.

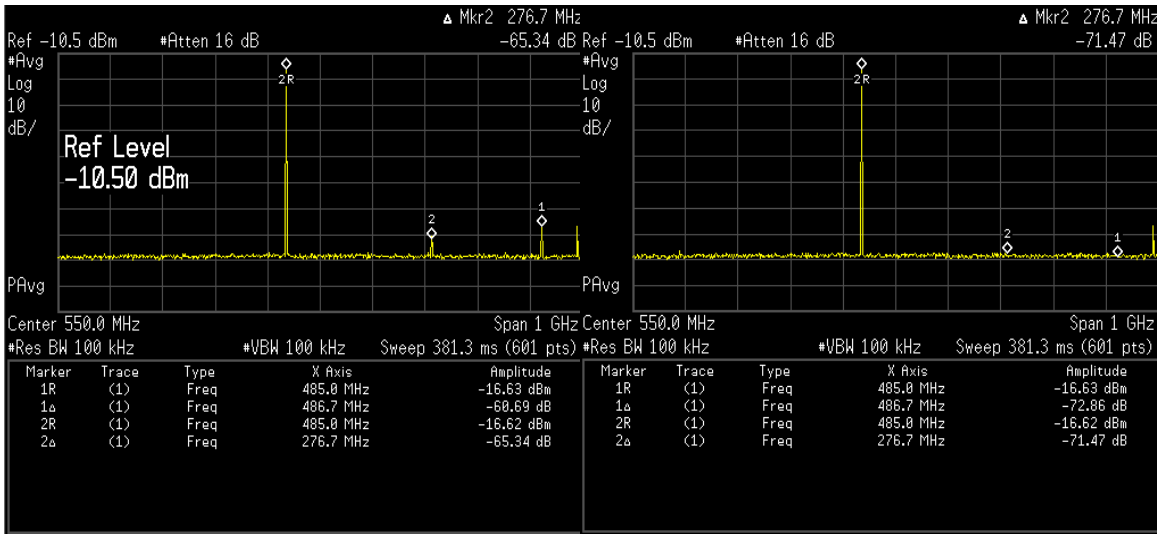


Figure 3.10: Distortion Cancellation for $F_c = 486$MHz

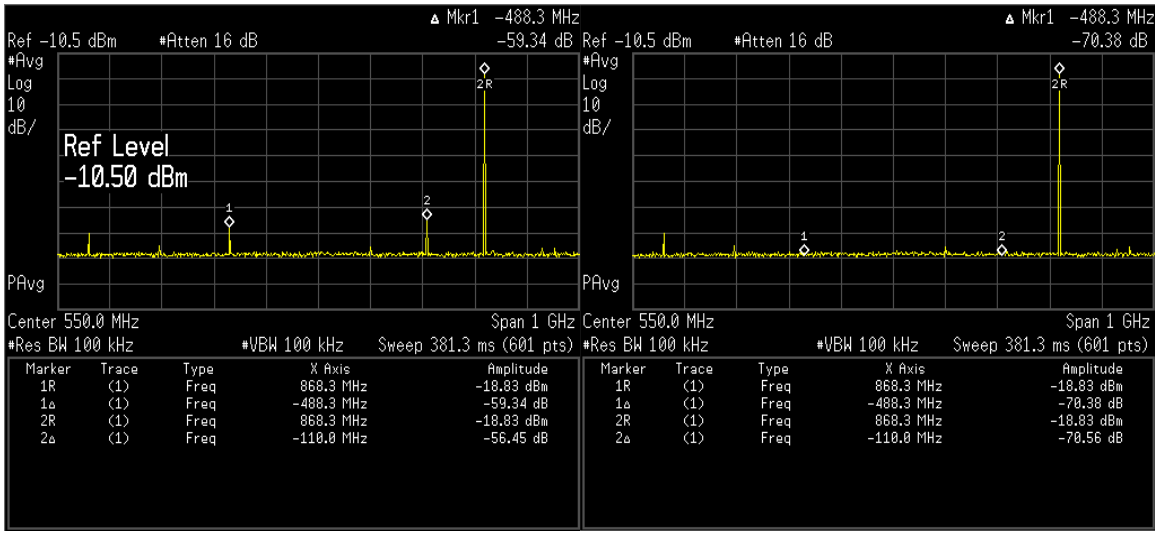


Figure 3.11: Distortion Cancellation for $F_c = 869$MHz

4. Designing a System to Cancel the Distortion Products

4.1 Overview

4.1.1 Matching the measured response

The model, measurements, and cancellation algorithm developed in Chapter 3 lead to the conclusion that the 2 distortion products can be cancelled by subtracting system that approximates the distortion frequency response from the desired signal. The system that does the processing can be modelled by a digital filtering structure. The list of pre-distortion architectures discussed in [27] suggest by omission that this is a novel approach.

Ideally a single filter can be used for each distortion product. If this method does not result in an acceptable response other filtering strategies could be used. Other options include sub-dividing the band or using a combination of two filters to separately approximate the amplitude and phase response.

4.1.2 Pre-distortion by Digital Filtering

A structure similar to Figure 4.1 will be used in a first attempt to cancel the distortion products.

The frequency response of each distortion product is to be approximated using an individual digital filter. The desired undistorted signal is used as an input for both of the cancellation filters. Since the frequency of the cancellation signal ends up being different than that of the output signal, the cancellation signal will not interfere with the spectrum of the desired signal. No operation is performed on the actual desired signal so there is no

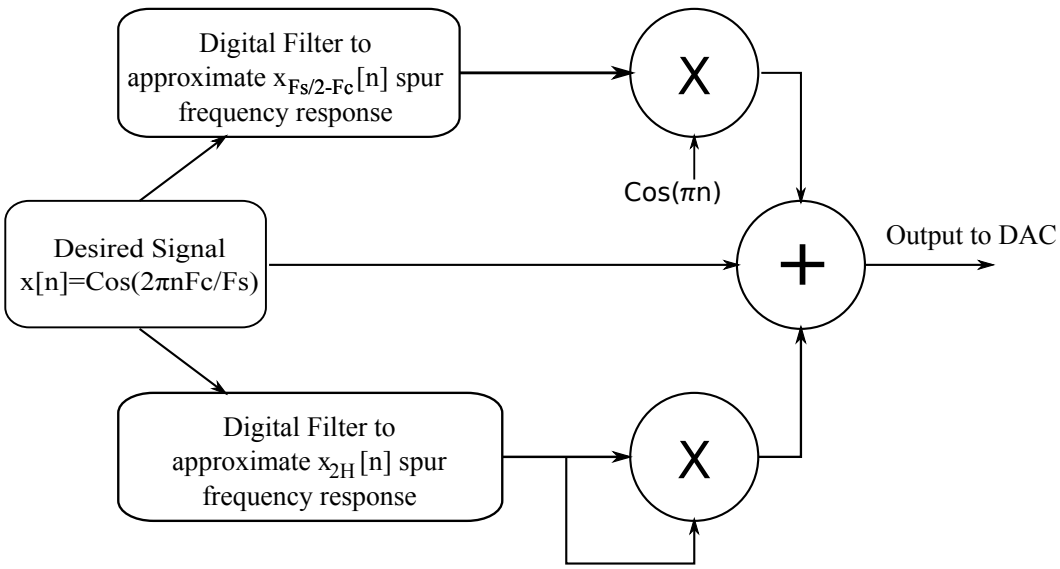


Figure 4.1: Block Diagram of Pre-distortion

danger of it being distorted or errors being introduced.

4.2 Using a Least Squared Error Algorithm to Generate a FIR filter

There are several algorithms that may be used to find a filter with a frequency response that tries to match the amplitude and phase response of a set of frequencies. Using an FIR filter for this purpose has advantages from a system integration and stability standpoint. An algorithm that minimizes the squared error between the filter and the measured data is a good approach in this application since all of the data points in the frequency band are of equal importance.

When using a least squared error method to match a desired, the fit can be improved over different segments of the band by increasing the number of data points in those segment. Candidate segments for increasing the data points are sub-bands where the spurs have a greater amplitude or where there is a high rate of change in amplitude or phase. The end result is more accurate curve fitting between points in the emphasized region of the spectrum.

4.2.1 Mathematical Basis

The mathematical basis for being able to generate FIR filter coefficients using a Least Squared Error method is detailed below. Let ω be a column vector of length N containing distinct frequency points with units of radians / sample

$$\omega = \begin{bmatrix} \omega_0, & \omega_1, & \cdots, & \omega_{N-1} \end{bmatrix}^T$$

Let h be a column vector of length M containing the coefficients of a FIR filter (coefficients are real)

$$h = \begin{bmatrix} b_0, & b_1, & \cdots, b_{M-1} \end{bmatrix}^T$$

and A be the N by M matrix given by

$$\mathrm{A} = e^{(j\omega \cdot [0:M-1])}. \tag{4.1}$$

Now the frequency response of filter h, denoted $H(e^{j\omega})$, has a value $H(e^{j\omega_k})$ at frequency $\omega = \omega_k$. Let H be a length N column vector containing the response for the frequencies in vector ω

$$H = \begin{bmatrix} H(e^{j\omega_0}) \\ H(e^{j\omega_1}) \\ \vdots \\ H(e^{j\omega_{N-1}}) \end{bmatrix} = Ah. \tag{4.2}$$

The problem is to find an h with a frequency response that is a "best fit" to a desired response at the frequencies in ω. The desired frequency response is denoted $H_D(e^{j\omega})$ and the desired responses at the frequencies in ω are denoted by vector H_D defined as

$$H_D = \begin{bmatrix} H_D(e^{j\omega_0}) \\ H_D(e^{j\omega_1}) \\ \vdots \\ H_D(e^{j\omega_{N-1}}) \end{bmatrix}. \tag{4.3}$$

In this case the "best fit" means the h that provides minimum mean squared error (MSE). The error, which is a complex column vector of length N is given by

$$E = H_D - H. \tag{4.4}$$

The scaler given by $E^H E$, where H used as a superscript indicates the conjugate transpose is given by

$$E^H E = (H_D - H)^H (H_D - H). \tag{4.5}$$

In other words $E^H E$ is the real number given by

$$E^H E = \sum_{k=0}^{N-1} |H_D e^{j\omega_k} - H e^{j\omega_k}|^2. \tag{4.6}$$

The problem is to find the vector h that minimizes $E^H E$. First express $E^H E$ as a function of h

$$\begin{aligned} E^H E &= (H_D - H)^H (H_D - H) \\ &= H_D^H H_D - H_D^H H - H^H H_D + H^H H \\ &= H_D^H H_D - H_D^H A h - h^H A_D^H + h^H A^H A h. \end{aligned} \tag{4.7}$$

Since h has M elements, which are $b_0, b_1, \cdots, b_{M-1}$; the optimum h is found by the simultaneous solution of the M equations

$$\begin{aligned} \frac{\partial E^H E}{\partial b_0} &= 0 \\ \frac{\partial E^H E}{\partial b_1} &= 0 \\ &\vdots \\ \frac{\partial E^H E}{\partial b_{M-1}} &= 0. \end{aligned}$$

The solution to these equations involves a vector of length M with a single non-zero element of 1. To make the analysis more compact the vector $\vec{1_k}$ is defined as a column vector with the $K+1^{th}$ entry being 1 and all others being 0. E.g. for $M = 5, K = 1, \vec{1_1} = [\ 0\ 1\ 0\ 0\ 0\ ]^T$. Since

$$\frac{\partial b_i}{\partial b_k} = \begin{cases} 0 & i \neq k \\ 1 & i = k \end{cases} \tag{4.8}$$

$$\frac{\partial h}{\partial b_k} = \begin{bmatrix} \frac{\partial b_0}{\partial b_k} \\ \frac{\partial b_1}{\partial b_k} \\ \vdots \\ \frac{\partial b_{M-1}}{\partial b_k} \end{bmatrix} = \vec{1_k}, \tag{4.9}$$

the partial derivative of $E^H E$ w.r.t. b_k is obtained from (4.7) and given by

$$\frac{\partial E^H E}{\partial b_k} = -H_D^H A\vec{1_k} - \vec{1_k}^H A^H H_D + \vec{1_k}^H A^H A h + h^H A H A\vec{1_k}. \tag{4.10}$$

Each of the terms on the right hand side reduces to a single element vector. Since the first term is the conjugate transpose of the second, these two single element vectors must be complex conjugates of each other, therefore

$$-H_D^H A\vec{1_K} - \vec{1_K} A^H H_D = -2\Re\left\{\vec{1_K}^H A^H H_D\right\}. \tag{4.11}$$

$\vec{1_K}^H$ is real and can be taken outside the "real" operator and expressed as $\vec{1_K}^T$. The expression becomes

$$-H_D^H A\vec{1_K} - \vec{1_K} A^H H_D = -2 \cdot \vec{1_K}^T \Re\left\{A^H H_D\right\}. \tag{4.12}$$

Similarly the last two terms can be expressed as

$$\vec{1_k}^H A^H A h + h^H A H A\vec{1_k} = 2\Re\left\{\vec{1_K}^H A^H A h\right\}, \tag{4.13}$$

and since both $\vec{1_k}^H$ and h are real (4.13) can be expressed as

$$\vec{1_k}^H A^H A h + h^H A H A\vec{1_k} = 2 \cdot \vec{1_K}^T \Re\left\{A^H A\right\} h. \tag{4.14}$$

The partial derivative can now be expressed as

$$\frac{\partial E^H E}{\partial b_k} = 2 \cdot \vec{1_K}^T (\Re\left\{A^H A\right\} h - \Re\left\{A^H H_D\right\}). \tag{4.15}$$

The M equations given by $\frac{\partial E^H E}{\partial b_k}$ can be expressed in matrix form. To demonstrate this it is convenient to view $\Re\{A^H A\}h - \Re\{A^H H_D\}$ as an M by 1 vector meaning a column vector, which is denoted Y. The column vector containing the partial derivatives can now be written in terms of Y

$$\begin{bmatrix} \frac{\partial E^H E}{\partial b_0} \\ \frac{\partial E^H E}{\partial b_1} \\ \vdots \\ \frac{\partial E^H E}{\partial b_{M-1}} \end{bmatrix} = \begin{bmatrix} 2 \cdot \vec{1_0}^T Y \\ 2 \cdot \vec{1_1}^T Y \\ \vdots \\ 2 \cdot \vec{1_{M-1}}^T Y \end{bmatrix} \tag{4.16}$$

$$= 2Y, \tag{4.17}$$

therefore

$$\frac{\partial E^H E}{\partial h} = \begin{bmatrix} \frac{\partial E^H E}{\partial b_0} \\ \frac{\partial E^H E}{\partial b_1} \\ \vdots \\ \frac{\partial E^H E}{\partial b_{M-1}} \end{bmatrix} = 2\Re\left\{A^H A\right\} h - 2\Re\left\{A^H H_D\right\}. \tag{4.18}$$

Setting $\frac{\partial E^H E}{\partial h}$ to zero provides a matrix equation for h

$$\Re\left\{A^H A\right\} h = \Re\left\{A^H H_D\right\}, \tag{4.19}$$

whose solution is

$$h = \Re\left\{A^H A\right\})^{-1} \Re\left\{A^H H_D\right\}. \tag{4.20}$$

This is the h that minimizes $E^H E$ and is referred to as h_m. The frequency response of the vector of coefficients given by h_m is designated $H_m(e^{j\omega})$.

4.2.2 Testing the Least Squares Method

Before attempting to match the measured frequency response this method is tested using two different FIR filters. The first checks to see if the formula is working correctly. It is a filter with the simple impulse response of [1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]. A Matlab script is used for the following tasks:

- Take the frequency response of this simple filter at fixed intervals over the frequency band.

- Construct the A matrix given by (4.1).
- Insert the frequency response vector and the A matrix into Equation(4.20) to re-generate the filter coefficients. In this case the filter coefficients are in fact the impulse response since the filter is an FIR filter.

If the algorithm is working correctly the coefficients returned for the filter should be exactly the same as the ones used to find the frequency response. This is assuming that the number of filter coefficients equals the number of data points given. Since our frequency response is generated by known coefficients the resultant vector should equal eleven 1's. The first trial indicates that this is indeed the case.

The formula also works if there are more data points than coefficients. Used in this way the calculation results in filter coefficients whose frequency response matches our data as well as possible by minimizing the least squared error between the resultant response and the desired response.

One other test filter, a high pass filter of order 21 is used. This filter does not have such simple coefficients and provides much more confidence that the algorithm will work in a real application. The Filter Design and Analysis tool (FDAtool) in Matlab is used to determine the coefficients for a more practical filter, resulting in the impulse response in Figure 4.2.

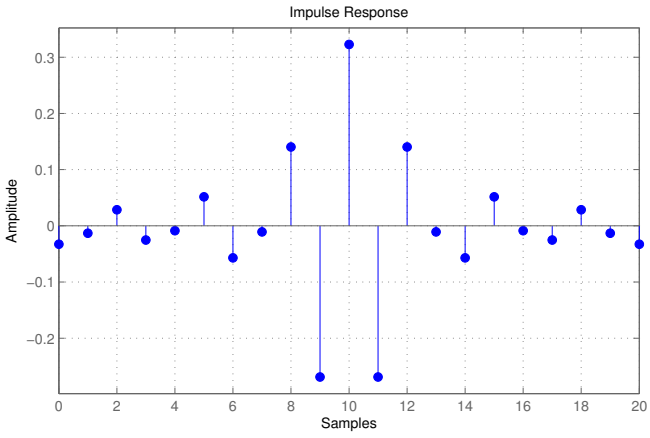
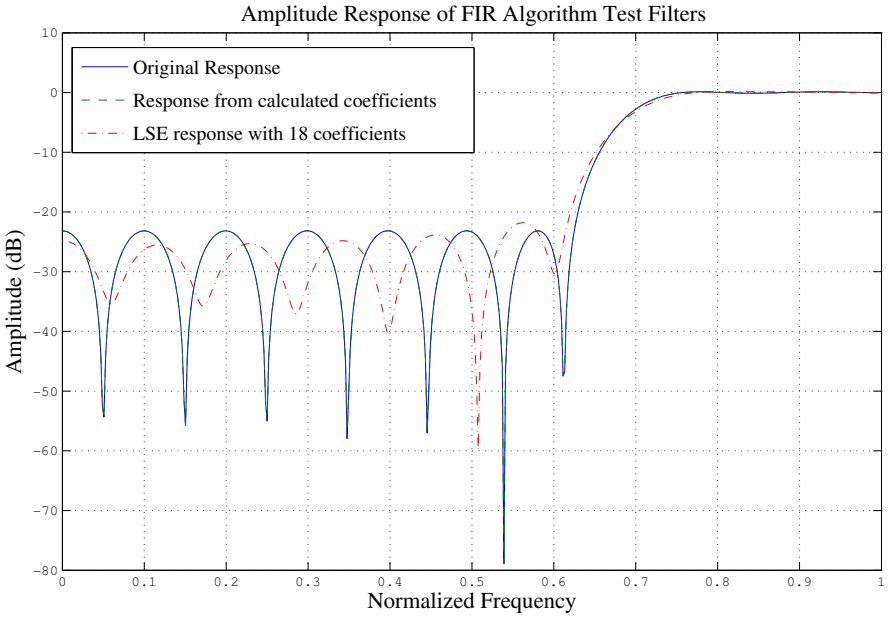


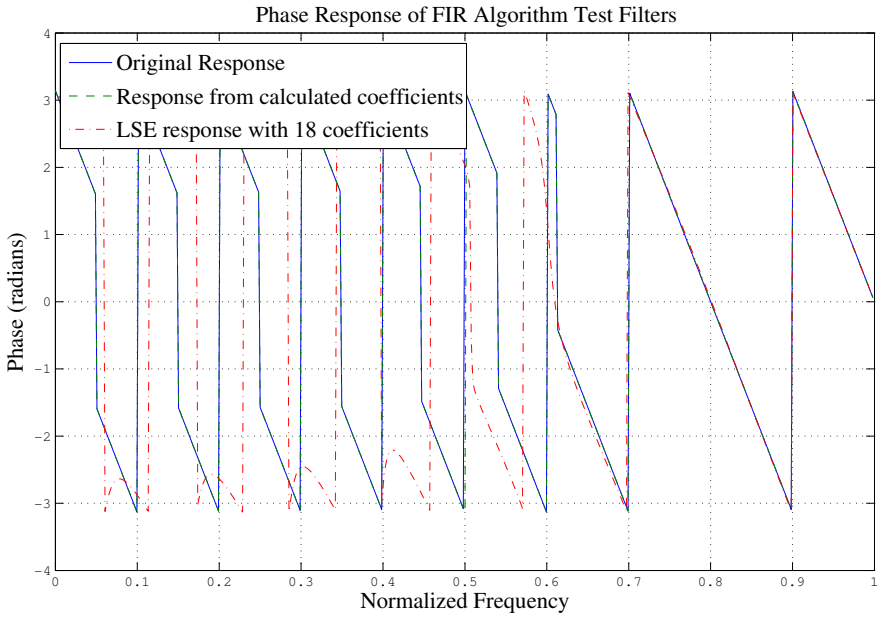
Figure 4.2: Impulse Response of High Pass test Filter

Again the resultant coefficients are exactly the same as the ones used to generate the frequency response. The amplitude and phase plots of the original filter as well as the filter with calculated coefficients are plotted in Figure 4.3 and found to be identical.

To check the least squared error performance of the algorithm, the frequency response of a filter that only uses 18 coefficients is included as well. The response shown in red in Figure 4.3 is clearly an approximation and there is a definite deviation in both amplitude and phase from the original filter.



(a) Amplitde

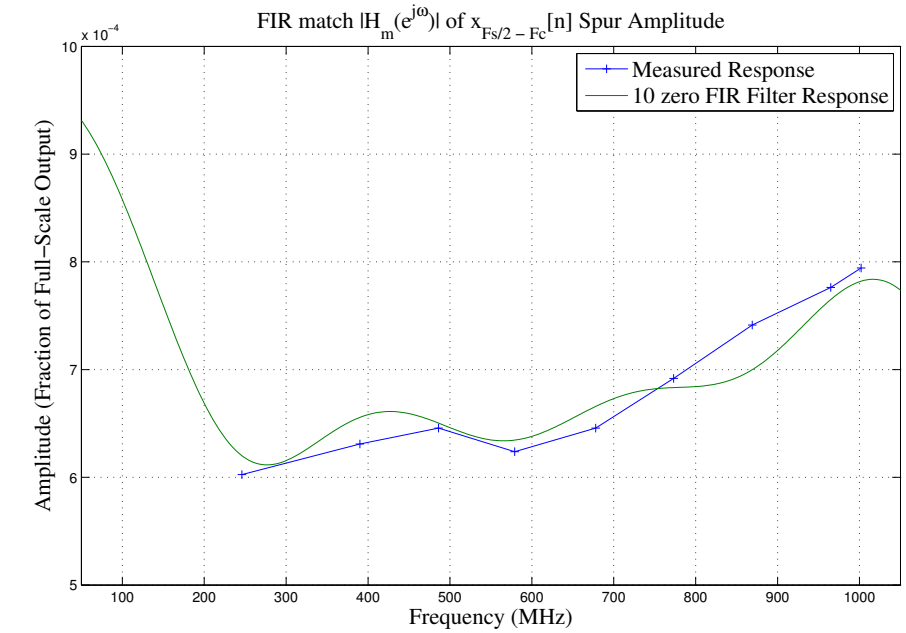


(b) Phase

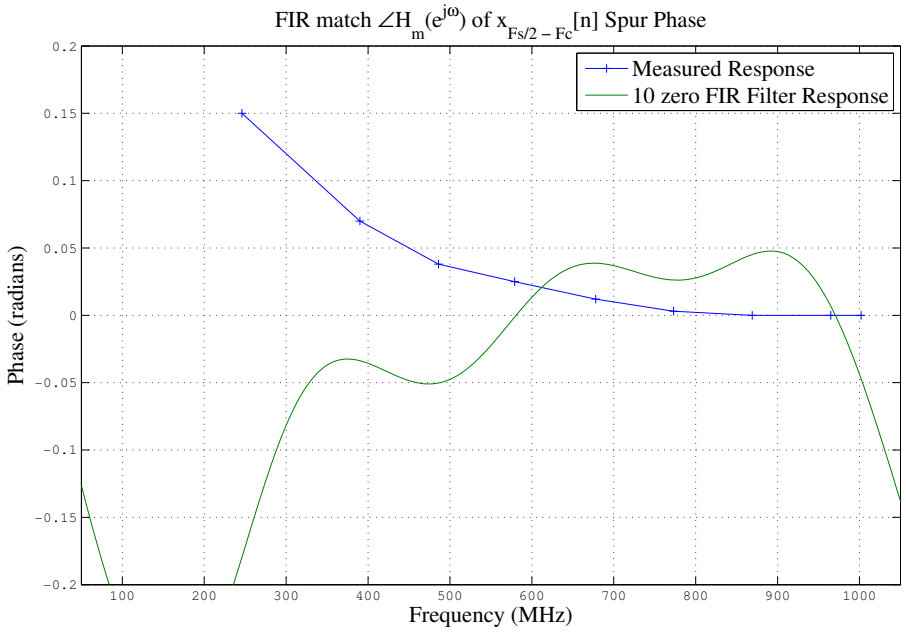
Figure 4.3: Amplitude and Phase response of original, calculated and LSE approx. of N=21 High Pass Filter

4.2.3 FIR Least Squares Method to Match the Distortion Frequency Response

After these tests, Equation 4.20 is used with the measured data points to generate filter coefficients that match the frequency responses for $x_{2H}[n]$ and $x_{F_s/2-F_c}[n]$. If the response is close enough this model may be used in a pre-distortion test system. The frequency responses of the filters are shown in Figures 4.4 and 4.5 along with the measured data.

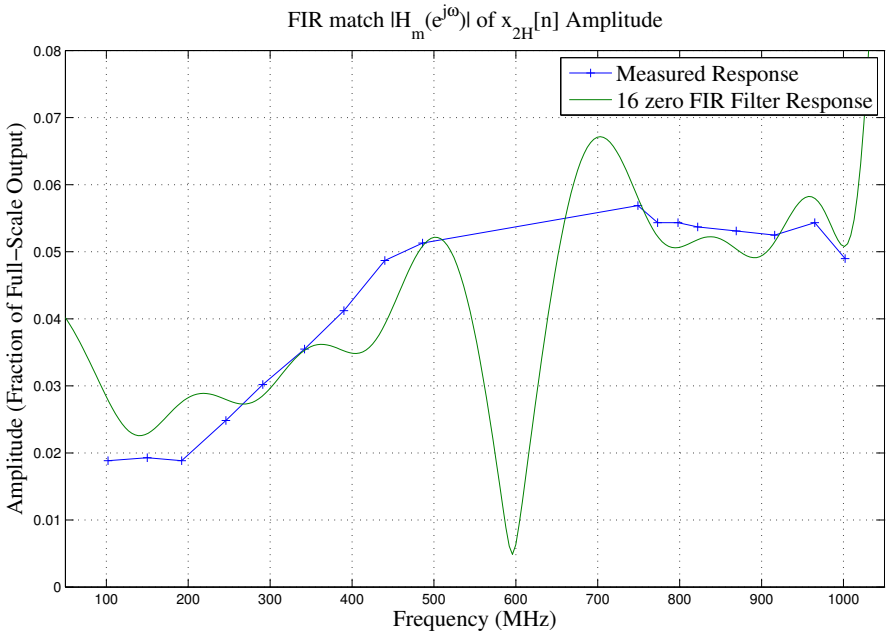


(a) Amplitde

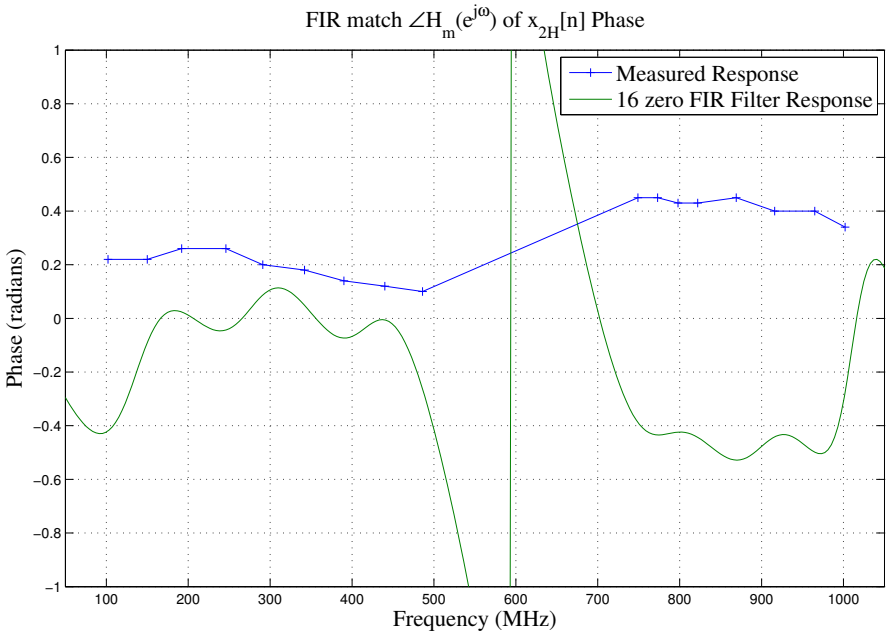


(b) Phase

Figure 4.4: Amplitude and Phase response of FIR filter to match $x_{F_s/2-F_c}[n]$



(a) Amplitde



(b) Phase

Figure 4.5: Amplitude and Phase Response of FIR Filter to Match $x_{2H}[n]$

Analysis of Results

In general the amplitude response of the filter can be matched quite well but the phase response is significantly different from what is desired. This difference in the phase response will not reduce the distortion products enough to make this a viable solution.

The most obvious observation leading to this conclusion is that for $x_{2H}[n]$ the phase change in the upper reaches of the band is actually opposite of what we need to reduce the spur and the two signals actually add constructively. Similarly for $x_{F_s/2-F_c}[n]$ the phase change is in the opposite direction to what is desired at the lower frequency ranges. This issue could be designed around by adding instead of subtracting the cancellation signal at the appropriate frequencies but again the frequency response would have to be divided into sub-bands. If the cancellation algorithm is being divided into sub-bands, the filters should be designed over the sub-band.

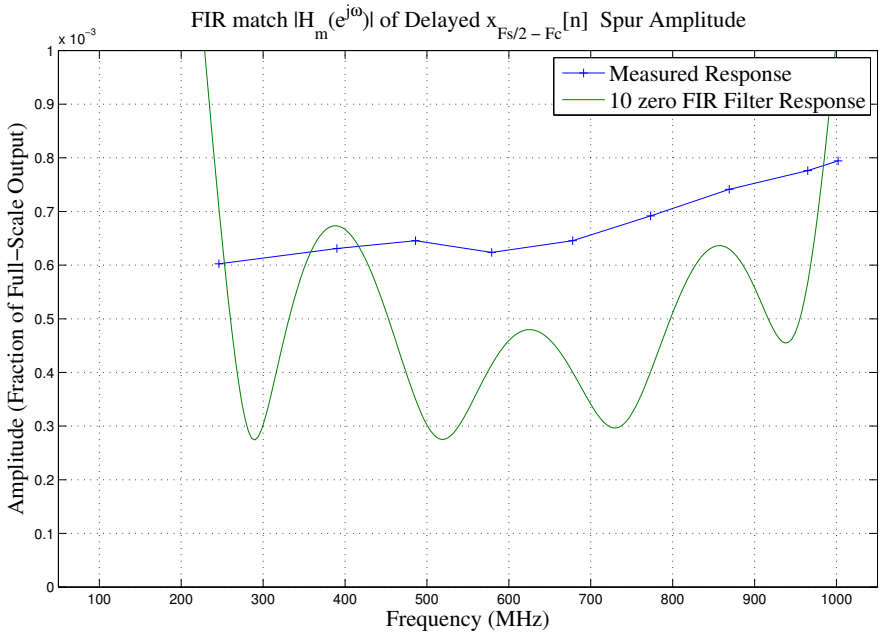
The fundamental problem is that in general an FIR filter structure will always result in the output being delayed compared to the input, and the higher frequency components of a signal are delayed to a larger degree than lower frequencies. The desired phase response is not close to a linear phase response. In the case of the filter to match $x_{F_s/2-F_c}[n]$, the desired phase response is close to 0 radians for the upper third of the frequency band. The required response to match $x_{2H}[n]$ has less delay in the upper frequency range than the lower. Both of these esponses are different from the expected response of an FIR filter.

4.2.4 FIR Least Squares Match of Delayed Distortion Frequency Response

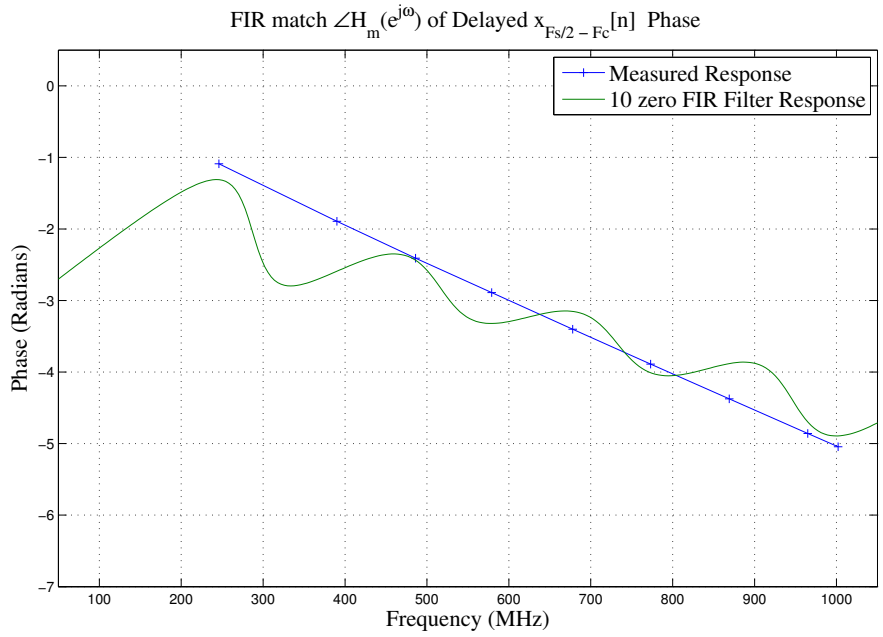
Since the main issue with using FIR filters is matching the phase component of the distortion products frequency response, perhaps the phase response can be altered. Adding a single delay register will add a linear phase change over frequency. The phase change is from 0 radians to π radians over the frequency range from 0 to $F_s/2$ for each delay register added. Equation 4.20 is again used with the delayed response.

The frequency responses of the 2 filters are shown in Figures 4.6 and 4.7. Various amounts

of delay were tested experimentally and good results were obtained when the response of $x_{F_s/2-F_c}[n]$ is delayed by 2 samples and $x_{2H}[n]$ is delayed by only a single sample.

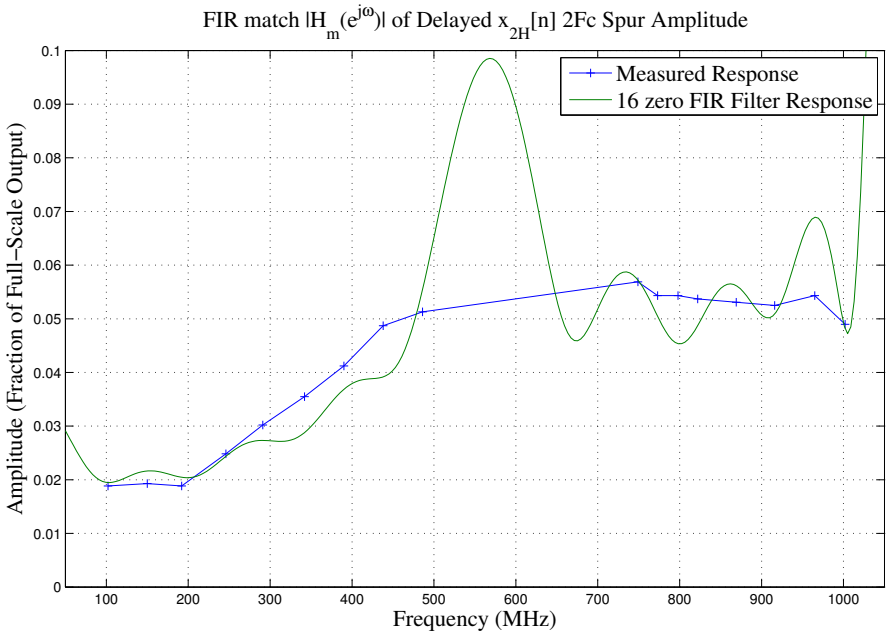


(a) Amplitde

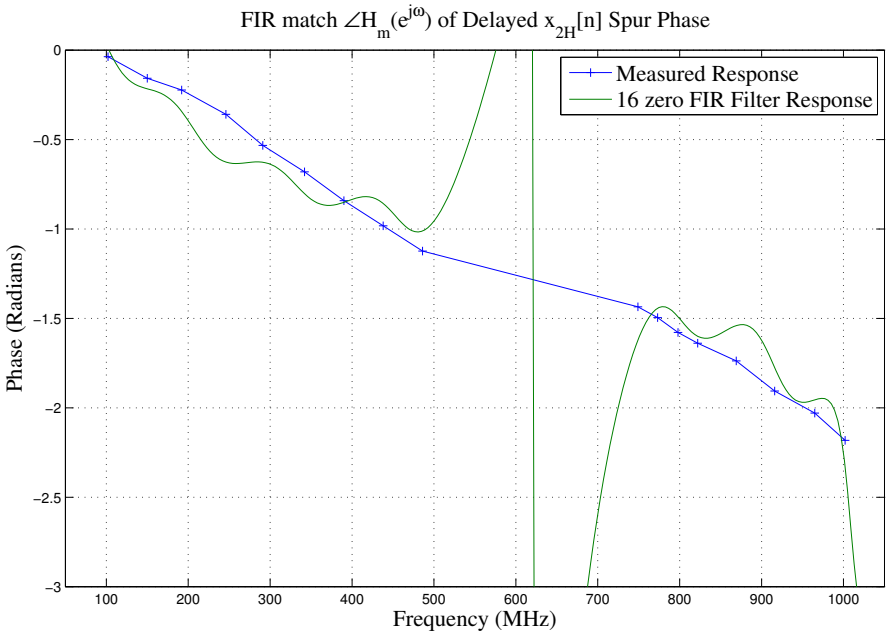


(b) Phase

Figure 4.6: Amplitude and Phase Response of FIR Filter to Match Delayed $x_{F_s/2-F_c}[n]$



(a) Amplitde



(b) Phase

Figure 4.7: Amplitude and Phase Response of FIR Filter to Match Delayed $x_{2H}[n]$

Analysis of $x_{F_s/2-F_c}[n]$ Distortion Match

The filter response match is somewhat improved when $x_{F_s/2-F_c}[n]$ is delayed. The expected attenuation at the measured data points is given in Table 4.1

Table 4.1: Calculated Attenuation of Delayed $x_{F_s/2-F_c}(t)$ at Datapoints

Frequency (MHz)	Calculated Attenuation (dBc)
246	10.58
390	2.98
486	6.78
579	6.30
678	7.38
773	7.32
869	6.48
965	10.45
1002	8.55

The expected attenuation is calculated by finding the power difference between a sinusoid with an amplitude and phase given by the measured data point, and the same sinusoid after the pre-distortion signal has been subtracted from it.

There are variations of this method that could be used to improve performance. The biggest challenge is to flatten the ripple in the response observed in Figure 4.6. To do this the order of the filter would have to be increased. Increasing the order can be done by either adding data points, or altering Equation 4.20 so the matrix does not have to be square. While these things are possible increasing the filter order is expensive from an FPGA resources standpoint, so other options will be investigated.

Analysis of $x_{2H}[n]$ Distortion Match

The filter performs dramatically better for the $x_{2H}[n]$ spur. The expected attenuation at the measured data points is given in Table 4.2

Table 4.2: Calculated Attenuation of Delayed $x_{2H}(t)$ at Datapoints

Frequency (MHz)	Calculated Attenuation (dBc)
102	24.48
150	17.23
192	16.37
246	11.56
291	17.66
342	13.22
390	19.47
440	12.93
486	16.98
749	14.56
773	21.23
798	14.73
822	20.02
869	13.53
916	17.58
965	11.02
1002	16.60

This version of the filter is certainly a candidate for use in a pre-distortion algorithm. The order would not have to be increased to test this filter. Since another filtering method has to be investigated to find a better match for $x_{F_s/2-F_c}[n]$, further testing with real signals will be done when both spurs can be cancelled at the same time.

4.3 A Different Algorithm Utilizing an IIR Filter Structure

Given the drawbacks with using an FIR filter to match the measured frequency response another filter option must be explored. Aside from FIR filters the other main style of digital filters are recursive filters, usually called Infinite Impulse Response or IIR filters. IIR filters

have an advantage over FIR filters in that a desired magnitude response can usually be achieved using lower order filters. The two main drawbacks are the non-flat group delay in the filter passband and the second is the potential for instability where the feedback inherent in the filter structure causes the output to become unstable.

Usually IIR filters are not used to filter signals that are digitally modulated because the phase distortion will corrupt any information that is encoded in the phase of the signal[1]. In this particular instance the IIR filters will not be used on the desired signal so there is no danger of the phase of the desired signal getting corrupted. The block diagram still matches Figure 4.1.

4.3.1 An Algorithm to find IIR Filter Coefficients that Match a Desired Response

In a search of existing literature one particular work titled "Algorithms for the Constrained Design of Filters with Arbitrary Magnitude and Phase Response" [28] stood out. Several algorithms are outlined to find filter coefficients, one used a variation of the Gauss-Newton algorithm to iteratively find a solution. The problem has not changed, find the minimum value of equation (4.6), the sum of the squared error at all the measured data points.

With an IIR filter there is a difference in the definition of the vector of coefficients h. h is still a column vector with N real elements, but now it includes the coefficients of both the numerator and denominator of the transfer function used to find the frequency response $H(e^{j\omega})$ at the M measured frequency points $(\omega_1, \omega_2, \ldots, \omega_M)$. A general form of h is

$$h = \Big[a_1, \quad a_2, \quad \cdots, a_P, \quad b_0, \quad b_1, \quad \cdots, b_Z\Big]^T. \tag{4.21}$$

P and Z are the numbers of coefficients in the denominator and numerator respectively and $N = P + Z$. Since $H(e^{j\omega})$ is dependent on h it can be re-written as $H(e_k^{j\omega}, h)$ and the sum

[1]There are some recursive filter structures that have a frequency response that is very close to linear phase over the passband.

of the error as

$$\sum_{k=1}^{M} \mathrm{w}_k |E_K|^2 = \sum_{k=1}^{M} \mathrm{w}_k |D_k - H(e^{j\omega_k}, h)|^2, \tag{4.22}$$

where E_K is a complex scaler indicating the error in the response at frequency ω_k. w_k is a real weighting factor assigned to $|E_K|^2$. The weighting factor simply assigns a relative importance to each data point compared to the others. The desired response is designated D_K. Now that there are poles in the transfer function, $H(e^{j\omega})$ has a non-linear dependency on the parameters in the vector h. To solve this non-linear system of equations 4 steps are taken.

1. Approximate the non-linear function of h, $H(e^{j\omega_k}, h)$, with a linear function, specifically a first order Taylor series expansion.

2. Using the linear approximation, find a quadratic equation that can be solved to find the parameter vector h that minimizes the mean square error in the approximation, i.e. minimize (4.22).

3. Set up the system of M linear equations,

$$\begin{bmatrix} E_1 \\ E_2 \\ \vdots \\ E_M \end{bmatrix} = A(h - h^{(0)}) \tag{4.23}$$

   where A is a square matrix of full rank and $h^{(0)}$ is the point where the Taylor series expansion is taken.

4. Solve for $(h - h^{(0)})$.

Expanding the 4 Steps to find a General Solution Using Linearized Equations

1. The approximate equivalent of $H = (e^{j\omega_k}, h)$ when linearized by a first order Taylor series approximation is given by

$$\begin{aligned} H(e^{j\omega_k}, h) &\approx \hat{H}(e^{j\omega_k}, h) \\ &= H(e^{j\omega_k}, h^{(0)}) + \nabla_h H(e^{j\omega_k}, h)\big|_{h=h^{(0)}} (h - h^{(0)}). \end{aligned} \tag{4.24}$$

$h^{(0)}$ is the designation for the initial value of h which is obtained by an estimation. ∇_h is a vector operator that takes the gradient of the scaler function. The ∇_h operator is the vector

$$\nabla_h = \left[\frac{\partial}{\partial h_1}, \frac{\partial}{\partial h_2}, \ldots, \frac{\partial}{\partial h_N}\right].$$

The approximate weighted error at each datapoint, $\sqrt{\mathrm{w}_k}\hat{E}_k$ is calculated using the equation

$$\sqrt{\mathrm{w}_k}\hat{E}_k = \sqrt{\mathrm{w}_k}(D_k - \hat{H}(e^{j\omega_k}, h)). \tag{4.25}$$

The linearized system (4.24) is then substituted into (4.25) resulting in the equation

$$\sqrt{\mathrm{w}_k}\hat{E}_k = \sqrt{\mathrm{w}_k}(D_k - H(e^{j\omega_k}, h^{(0)}) - \nabla_h H(e^{j\omega_k}, h)\big|_{h=h^{(0)}}(h - h^{(0)})). \tag{4.26}$$

To simplify the notation denote

$$\begin{aligned} D_k - H(e^{j\omega_k}, h^{(0)}) \quad &\text{by} \quad E_k^{(0)}, \text{ a scaler,} \\ \nabla_h H(e^{j\omega_k}, h)\big|_{h=h^{(0)}} \quad &\text{by} \quad \nabla_h H_k^{(0)}, \text{ a length } N \text{ row vector,} \\ h - h^{(0)} \quad &\text{by} \quad \delta^{(0)}, \text{ a length } N \text{ column vector,} \end{aligned}$$

Using this notation (4.25) becomes

$$\sqrt{\mathrm{w}_k}\hat{E}_k = \sqrt{\mathrm{w}_k}(E_k^{(0)} - \nabla_h H_k^{(0)}\delta^{(0)}). \tag{4.27}$$

2. Squaring the error in (4.27) and expanding the resulting quadratic equation gives the expression

$$\begin{aligned} \mathrm{w}_k|\hat{E}_k|^2 &= \mathrm{w}_k\left[(E_k^{(0)} - \nabla_h H_k^{(0)}\delta^{(0)})(E_k^{(0)} - \nabla_h H_k^{(0)}\delta^{(0)})^*\right] \\ &= \mathrm{w}_k\Big[|E_k^{(0)}|^2 - \nabla_h H_k^{(0)}\delta^{(0)}(E_k^{(0)})^* - E_k^{(0)}(\nabla_h H_k^{(0)}\delta^{(0)})^* \\ &\qquad + (\nabla_h H_k^{(0)}\delta^{(0)})(\nabla_h H_k^{(0)}\delta^{(0)})^*\Big]. \end{aligned} \tag{4.28}$$

Terms 2 and 3 in the square brackets are complex conjugates of each other. Furthermore since $\mathrm{w}_k, \nabla_h H_k^{(0)}\delta^{(0)}$, and $E_k^{(0)}$ are scalers and $\delta^{(0)}$ is real, the sum of these terms can be written as $2 \cdot \Re\{E_k^{(0)})^* \mathrm{w}_k(\nabla_h H_k^{(0)})\} \cdot \delta^{(0)}$. Term 4 of (4.28) is also the product of scalers and since $\nabla_h H_k^{(0)}\delta^{(0)}$ is equal to $(\delta^{(0)})^T(\nabla_h H_k^{(0)})^H$ the weighted squared error

reduces to

$$\begin{aligned} \mathrm{w}_k|\hat{E}_k|^2 = \mathrm{w}_k|E_k^{(0)}|^2 &- 2\cdot\Re\{E_k^{(0)})^*\mathrm{w}_k(\nabla_h H_k^{(0)})\}\cdot\delta^{(0)} \\ &+ (\delta^{(0)})^T(\nabla_h H_k^{(0)})^H\mathrm{w}_k(\nabla_h H_k^{(0)})(\delta^{(0)}). \end{aligned} \tag{4.29}$$

If the designation E_{Total} is used for the sum of the squared errors in (4.22) and the designation for the approximation of this sum using the linearized system is $\hat{E}_{Total}$, then the approximation of the sum of the squared error becomes

$$E_{Total} \approx \hat{E}_{Total} = \sum_{k=1}^{M} \mathrm{w}_k|\hat{E}_k|^2.$$

Substituting the weighted squared error at each datapoint given by (4.29) into the above summation allows $\hat{E}_{Total}$ to be expressed as

$$\begin{aligned} \hat{E}_{Total} &= \sum_{k=1}^{M} \mathrm{w}_k|E_k^{(0)}|^2 - 2\cdot\Re\left\{\sum_{k=1}^{M}(E_k^{(0)})^*\mathrm{w}_k(\nabla_h H_k^{(0)})\right\}\cdot\delta^{(0)} \\ &\quad +(\delta^{(0)})^T\left[\sum_{k=1}^{M}(\nabla_h H_k^{(0)})^H\mathrm{w}_k(\nabla_h H_k^{(0)})\right]\cdot\delta^{(0)}. \end{aligned} \tag{4.30}$$

Note that $\delta^{(0)}$ does not depend on k so it has been taken out of the summation.

3. Equation (4.30) can now be consolidated into matrix form using the following definitions. Let W be the $M \times M$ diagonal matrix

$$W = \begin{bmatrix} \mathrm{w}_1 & 0 & \cdots & 0 \\ 0 & \mathrm{w}_2 & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathrm{w}_M \end{bmatrix},$$

$E^{(0)}$ be the $1 \times M$ row vector

$$E^{(0)} = [E_1^{(0)}, E_2^{(0)}, \cdots, E_M^{(0)}],$$

and $\nabla_h H^{(0)}$ be the $M \times N$ matrix given by

$$\nabla_h H^{(0)} = \begin{bmatrix} \nabla_h H_1^{(0)} \\ \nabla_h H_2^{(0)} \\ \vdots \\ \nabla_h H_M^{(0)} \end{bmatrix}.$$

Using these definitions (4.30) can be written as

$$\begin{aligned} \hat{E}_{Total} = E^{(0)} W \left(E^{(0)}\right)^H &- 2 \cdot \Re\Big\{ \left(E^{(0)}\right)^* W (\nabla_h H^{(0)}) \Big\} \cdot \delta^{(0)} \\ &+ (\delta^{(0)})^T (\nabla_h H^{(0)})^H W (\nabla_h H^{(0)}) \cdot (\delta^{(0)}). \end{aligned} \tag{4.31}$$

To further consolidate the expression define

$$\begin{aligned} F^{(0)} &= 2 \cdot \Re\Big\{ \left(E^{(0)}\right)^* W (\nabla_h H^{(0)}) \Big\}, \text{ a } 1 \times N \text{ row vector} \\ A^{(0)} &= (\nabla_h H^{(0)})^H W (\nabla_h H^{(0)}), \text{ an } N \times N \text{ matrix,} \end{aligned}$$

then

$$\hat{E}_{Total} = E^{(0)} W \left(E^{(0)}\right)^H - F^{(0)} \delta^{(0)} + (\delta^{(0)})^T A^{(0)} \delta^{(0)}. \tag{4.32}$$

4. Find the value for $\delta^{(0)}$ that minimizes $\hat{E}_{Total}$. To do this solve the equations

$$\begin{aligned} \frac{\partial \hat{E}_{Total}}{\partial \delta_1^{(0)}} &= 0 \\ \frac{\partial \hat{E}_{Total}}{\partial \delta_2^{(0)}} &= 0 \\ &\vdots \\ \frac{\partial \hat{E}_{Total}}{\partial \delta_M^{(0)}} &= 0. \end{aligned} \tag{4.33}$$

If M is greater than or equal to N a single least squared error solution can be found for h.

Since the first term in (4.32) does not depend on $\delta_k^{(0)}$ the partial derivative of that term

is zero. To find the partial derivative of the second term of (4.32)

$$
\begin{aligned}
F^{(0)}\delta^{(0)} &= F^{(0)} \begin{bmatrix} \delta_1^{(0)} \\ \delta_2^{(0)} \\ \vdots \\ \delta_N^{(0)} \end{bmatrix} \\
\frac{\partial F^{(0)}\delta^{(0)}}{\partial \delta_k^{(0)}} &= F^{(0)} \begin{bmatrix} \frac{\partial \delta_1^{(0)}}{\partial \delta_k^{(0)}} \\ \frac{\partial \delta_2^{(0)}}{\partial \delta_k^{(0)}} \\ \vdots \\ \frac{\partial \delta_N^{(0)}}{\partial \delta_k^{(0)}} \end{bmatrix} \\
&= F^{(0)} \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix} \text{ where the 1 is at the k}^{th}\text{row.}
\end{aligned}
$$

Taking the vector product produces a scaler consisting of the kth element of $F^{(0)}$ denoted $F_k^{(0)}$, therefore

$$\frac{\partial F^{(0)}\delta^{(0)}}{\partial \delta_k^{(0)}} = F_k^{(0)}. \tag{4.34}$$

Finally the partial derivative of the third term in (4.32) is shown below

$$\frac{\partial((\delta^{(0)})^T A^{(0)}\delta^{(0)})}{\partial \delta_k^{(0)}} = \left(\frac{\partial(\delta^{(0)})^T}{\partial \delta_k^{(0)}}\right) A^{(0)}\delta_k^{(0)} + (\delta_k^{(0)})^T A^{(0)} \left(\frac{\partial \delta^{(0)}}{\partial \delta_k^{(0)}}\right)$$

By applying the same logic used for term two, the partial derivatives are reduced to a row and column vector with all elements equal to 0 except for a 1 at the k$^{\text{th}}$ element. Using this notation the third term becomes

$$\frac{\partial((\delta^{(0)})^T A^{(0)}\delta^{(0)})}{\partial\delta_k^{(0)}} = \left[0,\cdots,0,1,\cdots,0\right] A^{(0)}\delta^{(0)} + (\delta_k^{(0)})^T A^{(0)} \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix}.$$

Since

$$[0,\cdots,0,1,\cdots,0]A^{(0)} = \text{ row k of A}^{(0)},$$

and

$$A^{(0)} \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix} = \text{column } k \text{ of } A^{(0)},$$

the partial derivative reduces to

$$\frac{\partial((\delta^{(0)})^T A^{(0)}\delta^{(0)})}{\partial\delta_k^{(0)}} = A^{(0)}_{\text{row\_}k}\delta_k^{(0)} + (\delta(0))^T A^{(0)}_{\text{column\_}k}. \tag{4.35}$$

The product of $(\delta^{(0)})^T A^{(0)}_{\text{column\_}k}$ is a 1×1 matrix and must be equivalent to its transpose $(A^{(0)}_{\text{column\_}k})^T\delta^{(0)}$. This allows (4.35) to be written as

$$\frac{\partial((\delta^{(0)})^T A^{(0)}\delta^{(0)})}{\partial\delta_k^{(0)}} = A^{(0)}_{\text{row\_}k}\delta_k^{(0)} + (A^{(0)}_{\text{column\_}k})^T\delta(0).$$

Also since $A^{(0)} = (A^{(0)})^H$, $(A^{(0)}_{\text{column\_}k})^T$ must equal $(A^{(0)}_{\text{row\_}k})^*$, therefore

$$\begin{aligned}\frac{\partial((\delta^{(0)})^T A^{(0)}\delta^{(0)})}{\partial\delta_k^{(0)}} &= \left(A_{\text{row\_}k}^{(0)} + (A_{\text{row\_}k}^{(0)})^*\right)\delta^{(0)} \\ &= 2\cdot\Re\left\{A_{\text{row\_}k}^{(0)}\right\}\delta^{(0)}. \end{aligned} \tag{4.36}$$

Now the system of equations in (4.33) can be expressed in matrix form,

$$\frac{\partial\hat{E}_{Total}}{\partial\delta_k^{(0)}} = -F_{\text{element\_}k}^{(0)} + 2\cdot\Re\left\{A_{\text{row\_}k}^{(0)}\right\}$$

$$\begin{bmatrix} \frac{\partial\hat{E}_{Total}}{\partial\delta_1^{(0)}} \\ \frac{\partial\hat{E}_{Total}}{\partial\delta_2^{(0)}} \\ \vdots \\ \frac{\partial\hat{E}_{Total}}{\partial\delta_M^{(0)}} \end{bmatrix} = -F^{(0)} + 2\cdot\Re\left\{A^{(0)}\right\}\delta^{(0)}.$$

The next step is to set the partial derivatives to equal 0 and solve for $\delta^{(0)}$

$$\begin{aligned} 0 &= -F^{(0)} + 2\cdot\Re\left\{A^{(0)}\right\}\delta^{(0)} \\ F^{(0)} &= 2\cdot\Re\left\{A^{(0)}\right\}\delta^{(0)} \\ \delta^{(0)} &= \frac{1}{2}\left(\Re\left\{A^{(0)}\right\}\right)^{-1}F^{(0)} \end{aligned} \tag{4.37}$$

The value of h that minimizes $\hat{E}_{Total}$ is

$$h = h^{(0)} + \delta^{(0)} \tag{4.38}$$

The new value for h will be in significant error if $\delta^{(0)}$ is anything but infinitesimal because the linearized set of equations will have error. This means h must be calculated iteratively. Unfortunately this iterative algorithm can be unstable if $\delta^{(0)}$ is large. To compensate for this the calculated $\delta^{(0)}$ in (4.38) can be multiplied by a factor α, a real positive number less than 1. Making α small ensures stability by eliminating the overshoot that can occur when the $\nabla_h^{(0)}$ for $h = h_0$ is greater than the final solution.

Using the compensation factor α, the procedure can be repeated iteratively solving for small increments of $\delta^{(0)}$ until the sum of the error at the data points is very small. The resulting coefficient vector from using this method is defined as h_i and the frequency response defined as H_i.

4.3.2 Finding coefficients to match the response of a 2-pole filter

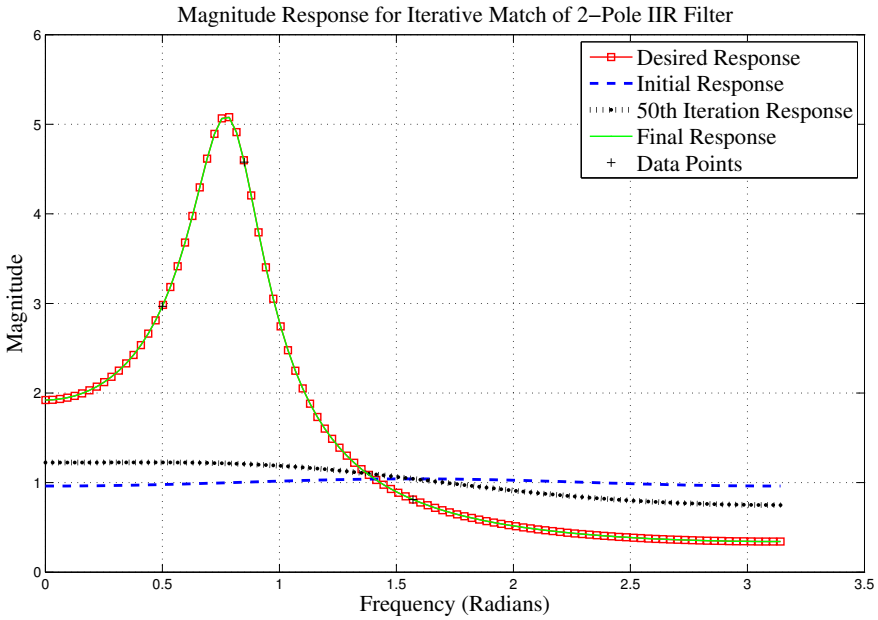
To demonstrate the method a 2-pole filter with the form is used.

$$H(e^{j\omega}, r, \theta) = \frac{1}{(1 - 2rcos(\theta)e^{-j\omega} + r^2e^{-2j\omega})} \tag{4.39}$$

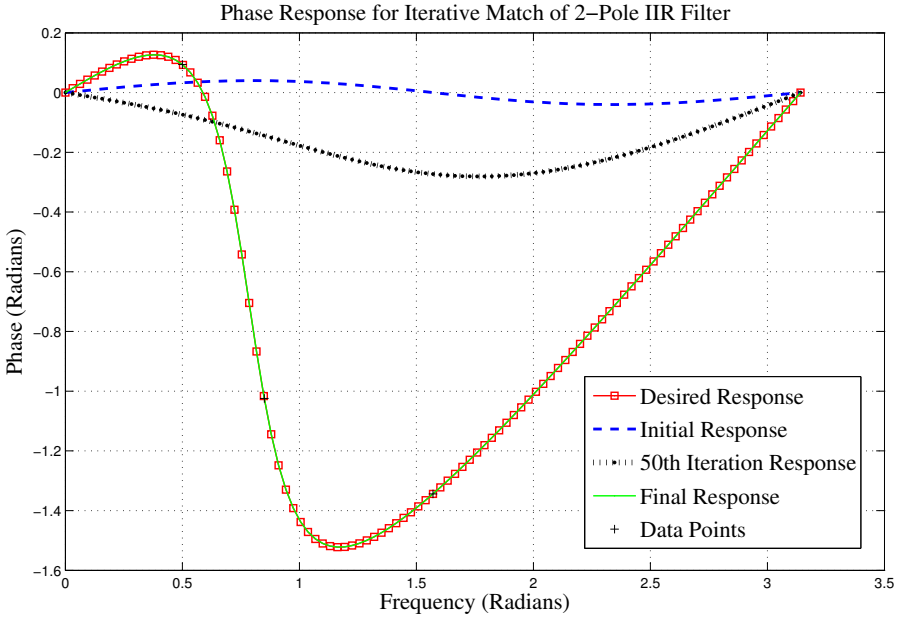
In this example the 2 parameters are the angle and radius of the 2 poles plotted in the z-plane. Since the filter has real coefficients the poles must be complex conjugates of each other. Alternately the parameter vector could be comprised of the 2 real coefficients of the filter.

The correct result for the parameters, selected for the purposed of this demonstration are at $r = 0.85$ and $\theta = \pi/4$. The starting values are chosen randomly to be $r = 0.2$ and $\theta = \pi/2$. To allow the algorithm to calculate the error, 3 frequency points, instead of the minimum of 2 at $\omega = 0.5, 0.85, \pi/2$ radians are used. The ∇ matrix is a 3x2 matrix that has each row containing the partial derivative of (4.39) with respect to each of the parameters, evaluated at one of the frequency points. The matrix is used to calculate vectors for $F^{(0)}$ and $A^{(0)}$, which in turn are used to calculate the δ values in equation (4.37) for the parameters. An α value of 0.01 is used and then the process is repeated until the sum of the errors at all data points is less than 0.005.

The results of this trial are excellent. The algorithm is able to match the data points exactly. This is not unexpected since the values at the selected frequencies are calculated using the "correct" parameters, and there should only be one unique solution for the parameters that will results in an exact match at the selected data points. Figure 4.8 shows the target filter, starting filter as well as the results after 50 iterations to show the progression of the algorithm.



(a) Amplitde

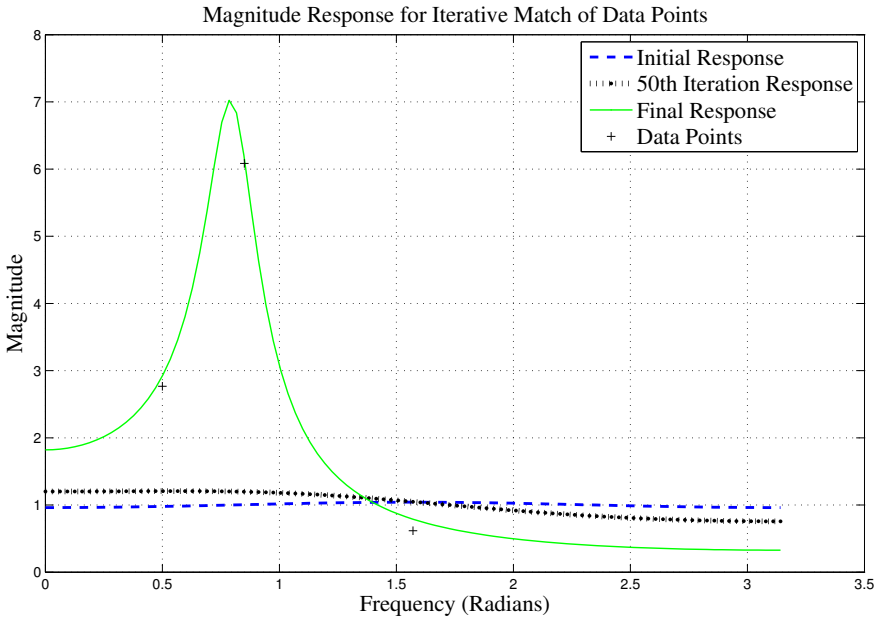


(b) Phase

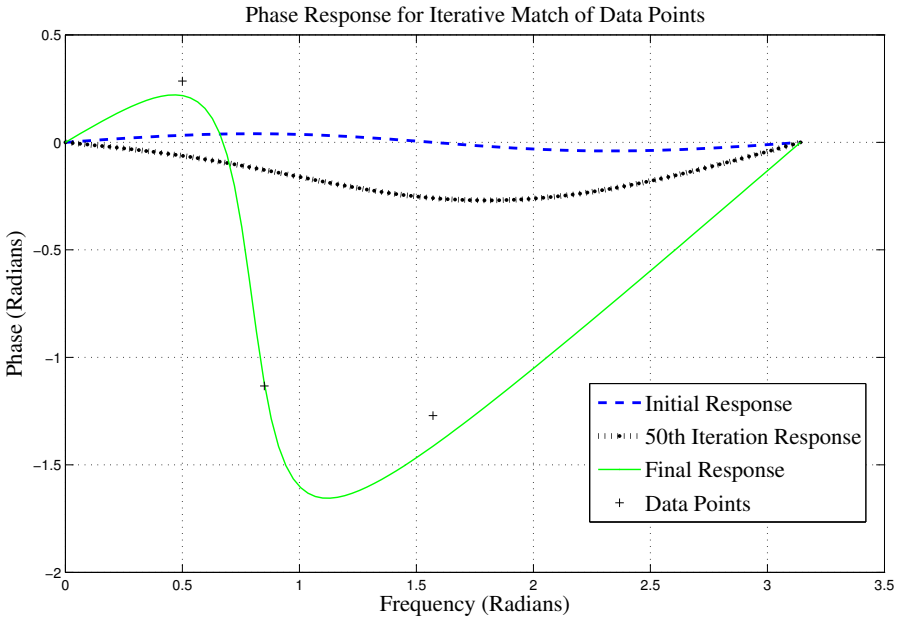
Figure 4.8: Amplitude and Phase Response of 2-pole Filter using Iterative Method

Note that the desired response and final result are indistinguishable from each other in the plots.

To give an example that more closely approximates matching measured data the target values are altered slightly. Now the data points can not be matched exactly but the algorithm stops the iterative procedure when the error is minimized as much as possible. The new data points, starting point, 50th iteration and final results are detailed in Figure 4.9.



(a) Amplitde



(b) Phase

Figure 4.9: Amplitude and Phase Response of 2-pole Filter using Iterative Method

4.3.3 Coefficients and Frequency Response of Generated by algorithm

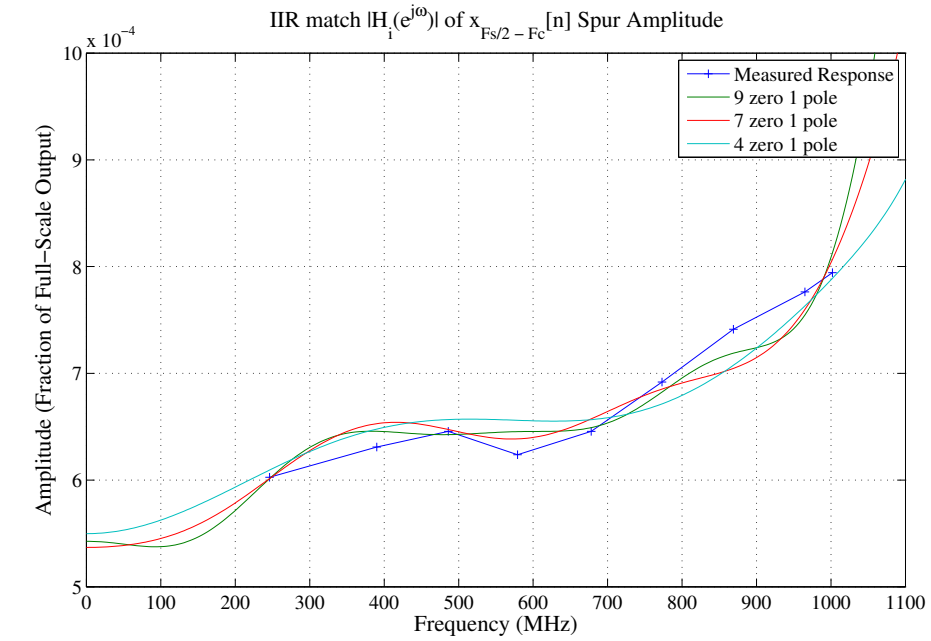
The algorithm to find the coefficients of IIR filters is now tested on the measured data points. The initial starting point in both cases is selected to be the coefficients found for an FIR filter. Initially a single pole is used. The initial value for the pole is selected to simply be 0.1.

Response of $x_{F_s/2-F_c}[n]$ Filter

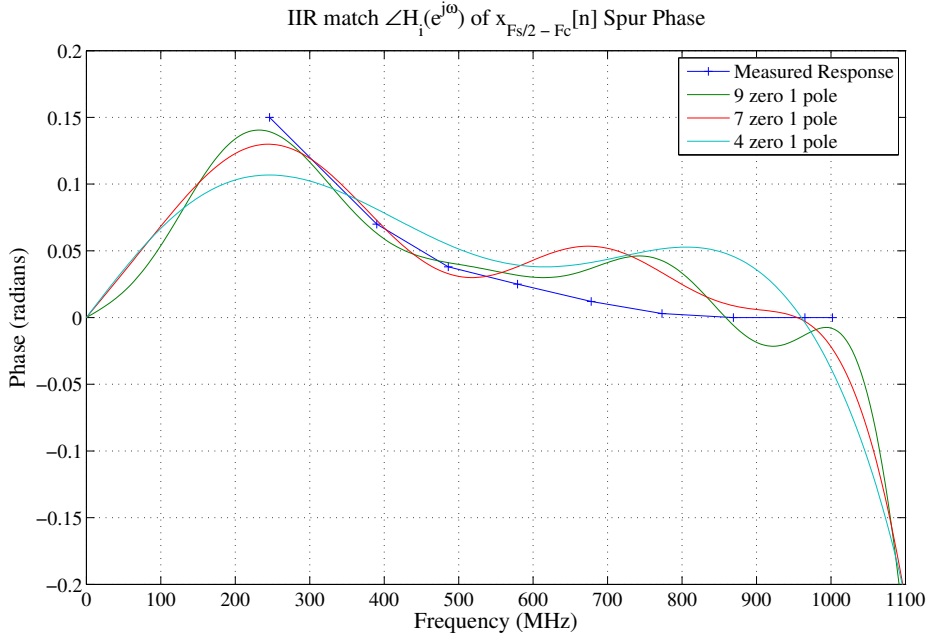
The results using the IIR algorithm show significant improvement compared to the non-delayed FIR filters especially in matching the measured phase response of the distortion signal. As part of the analysis the algorithm was tested with different numbers of zeros to compare how the frequency response matched the measured data as the order is increased.

The filter frequency response and measured data for $x_{F_s/2-F_c}[n]$ is shown in Figure 4.10. The response is shown for 9, 7 and 4 zeros. The progression of the match to the measured data is observable as the order is increased.

The poles and zeroes used in the filter are plotted in Figure 4.11. The results are good enough to proceed with developing a filter for this spur using the generated coefficients. Table 4.3 shows the expected attenuation of $x_{F_s/2-F_c}(t)$ at the measured data points.



(a) Amplitde



(b) Phase

Figure 4.10: Amplitude and Phase response of IIR filters to match $x_{F_s/2-F_c}[n]$

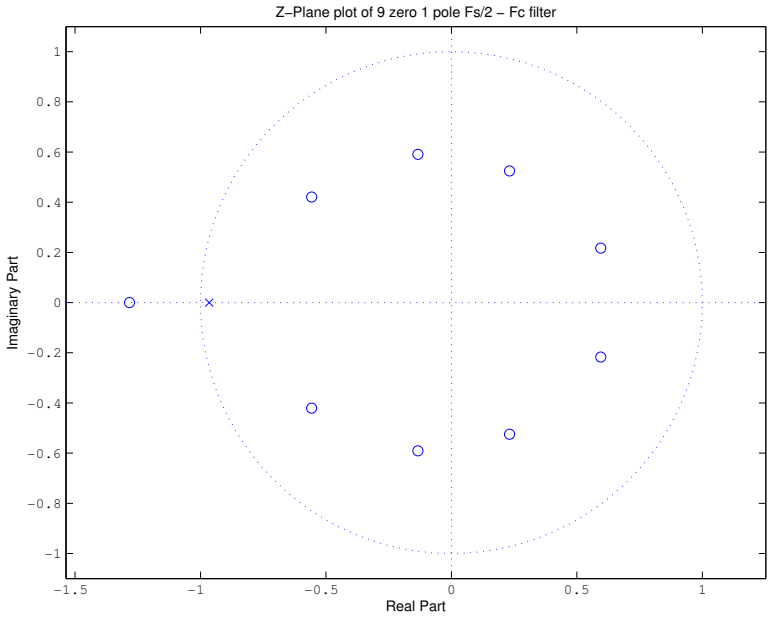


Figure 4.11: Z-plane plot of 9 zero 1 pole generated filter

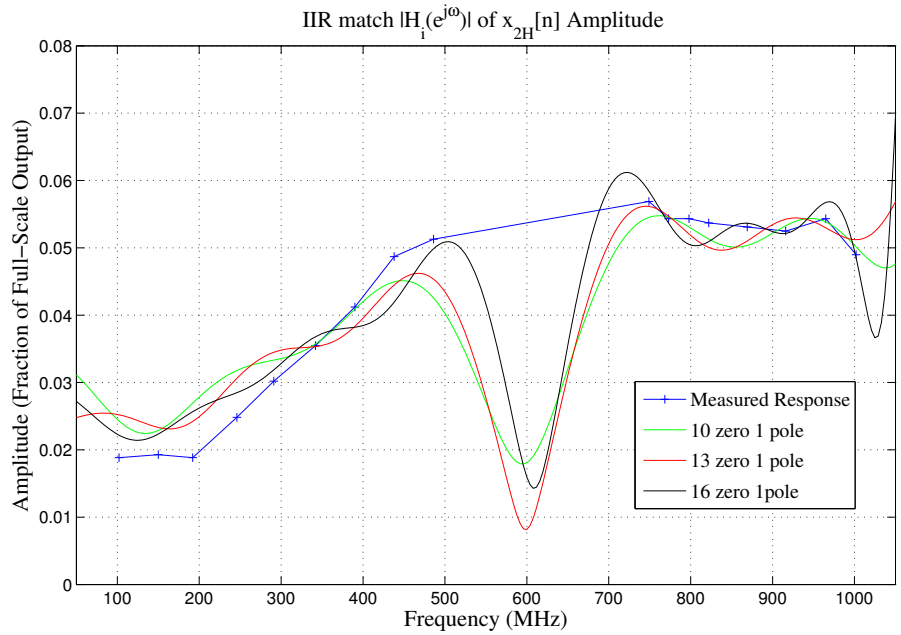
Table 4.3: IIR Calculated Attenuation of $x_{F_s/2-F_c}(t)$ Spur at Datapoints

Frequency (MHz)	Calculated Attenuation (dBc)
246	39.42
390	32.24
486	44.87
579	29.04
678	31.87
773	27.60
869	30.21
965	30.25
1002	32.17

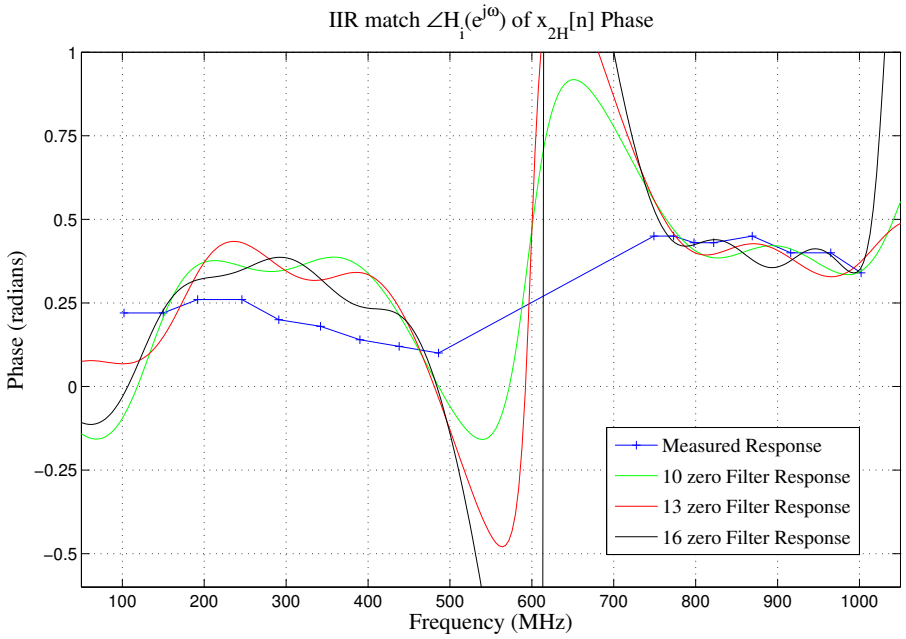
Response of $x_{2H}[n]$ Filter

The measured data for $x_{2H}[n]$ is now used as an input vector to the IIR algorithm. Again the results are very encouraging. There is a vast improvement compared to the FIR filter especially in the phase response. To observe the change in frequency response as the order of the filter changes Figure 4.12 plots the generated frequency response for 16, 13 and 10 zeros. The match of the frequency response is good enough that it is worth developing a filter based on these results.

Initially a single pole is used, but to try to improve the results, filters were tested with the number of poles increased from 1 to 3 and 6 while keeping a constant number of 16 zeros. Increasing the number of poles did not improve the results significantly enough to be worth the increase in filter cost and complexity.



(a) Amplitde



(b) Phase

Figure 4.12: Amplitude and Phase response of different order IIR filters to match $x_{2H}[n]$ spur

The poles and zeroes used in the filter are plotted in Figure 4.13.

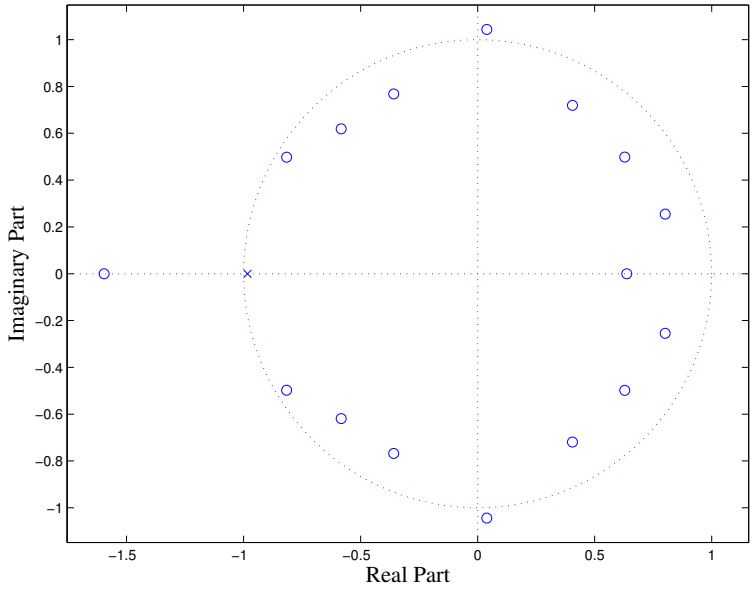


Figure 4.13: Z-plane plot of 16 zero 1 pole generated filter

Table 4.4 shows the expected attenuation of $x_{2H}(t)$ spur at the measured data points.

Table 4.4: IIR Calculated Attenuation of $x_{2H}(t)$ Spur at Datapoints

Frequency (MHz)	Calculated Attenuation (dBc)
102	9.99
150	15.95
192	8.67
246	15.13
291	14.01
342	16.54
390	18.40
440	14.25
486	18.64
749	21.33
773	33.73
798	23.04
822	25.60
869	22.87
916	32.21
965	26.90
1002	34.76

Analysis of initial results

The results in both the cases are good. Both the amplitude and the phase response match the measured results quite well. As for deciding what order of filter to use, increasing the number of zeros improves the match to the distortion signal while the benefits of increasing poles is somewhat minimal. For the test system the 9 zero 1 pole filter is used to cancel $x_{F_s/2-F_c}(t)$ while the 16 zero 1 pole filter is used to cancel $x_{2H}(t)$. In a real system the desired margin above the specification as well as the allocation of FPGA resources would be the factors determining the order of filter. Increasing the poles does improve the stability of the filter by moving the poles away from the unit circle so increasing the number of poles could be investigated if filter stability is an issue.

4.4 Building a Filter to Pre-distort a Real Signal

Now that filter coefficients have been determined, the next step is to build a filter in Matlab that will match a practical structure that could possibly be used to pre-distort the desired output signal. The filter is designed with a direct form 2 filter structure. Developing a filter this way allows for later analysis of quantizing noise and will greatly increase the ease with which the pre-distortion routine can be adapted for use in an FPGA based system.

```
function [dist_samp] = IIR_dist_fil_9z1p(X, coef_poly_z, coef_poly_p)
%Build the IIR filter to test the IIR filter algorithm
%Xn1, Xn2...etc. are the delay registers
%vn(x) registers are the results of the numberator coefficients
%gn(1) is the results of the denominator coefficient.
X = [X,X];
Y = ones(1, length(X));
V = ones(1, length(X));

for n = 1 : length(X);
    %Initialize the output vector and delay register
    if n == 1
        Y(n) = X(n);
```

```
    V(n) = X(n)* coef_poly_z(1);  %loading the output vector initially
    Xn1 = 0;
    Xn2 = 0;
    Xn3 = 0;
    Xn4 = 0;
    Xn5 = 0;
    Xn6 = 0;
    Xn7 = 0;
    Xn8 = 0;
    Xn9 = 0;
    Xn10 = 0;
else
    gn(1) = Xn1 * (-coef_poly_p(2));

    Y(n) = X(n) + sum(gn);

    Y_b0  = Y(n) * coef_poly_z(1);
    vn(1) = Xn1 * coef_poly_z(2);
    vn(2) = Xn2 * coef_poly_z(3);
    vn(3) = Xn3 * coef_poly_z(4);
    vn(4) = Xn4 * coef_poly_z(5);
    vn(5) = Xn5 * coef_poly_z(6);
    vn(6) = Xn6 * coef_poly_z(7);
    vn(7) = Xn7 * coef_poly_z(8);
    vn(8) = Xn8 * coef_poly_z(9);
    vn(9) = Xn9 * coef_poly_z(10);
    vn(10) = Xn10 * coef_poly_z(11);
    V(n) = (Y_b0 + sum(vn));

    Xn10 = Xn9;
    Xn9 = Xn8;
    Xn8 = Xn7;
    Xn7 = Xn6;
    Xn6 = Xn5;
    Xn5 = Xn4;
    Xn4 = Xn3;
```

```
        Xn3 = Xn2;
        Xn2 = Xn1;
        Xn1 = Y(n);
    end
%Remove the first half of the samples to let the filter settle
dist_samp = V(length(X)/2+1:length(X));
end
```

The Matlab function will take fixed length periodic sample streams and pass it through a digital filter. The above example is for the filter used to match the frequency response of $x_{F_s/2-F_c}[n]$. In order to allow for the filter to settle into a steady state the sample stream is concatenated with itself, looping through the samples twice, filtering them and then the second loop is returned as the output. All of the filters developed use this format. Figure 4.14 is a block diagram of the filter code.

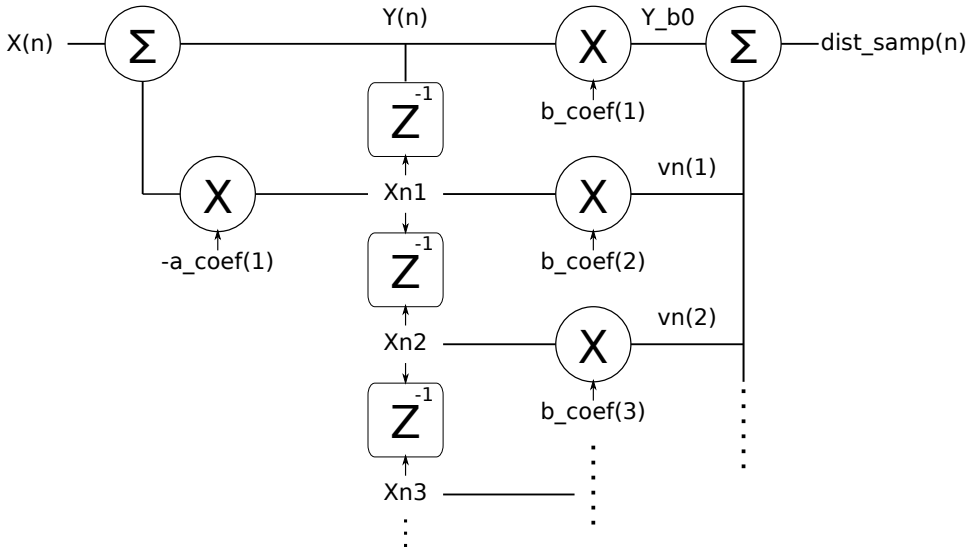


Figure 4.14: Block Diagram of Digital Filter Code

The input sample stream is denoted X with the sample taking the form of a vector $X = [\, X(1),\ X(2),\ X(3), \cdots, X(16384)\,]$.

The $x[n]$ generator is used to test out the filtering algorithm. The filter structure in Figure 4.14 is used for both of the filters in Figure 4.1. Each distortion signal sample is

multiplied by the appropriate factor (either π or squared) to match the frequency of the distortion product. The three signals are then combined on a sample by sample basis and then output to a text file that is uploaded into the DPG. Since a test tone is being pre-distorted, each of the cancellation is signals is created on its own and the resulting samples are stored in a vector. This structure is not suitable for a real-time signal but since the desired signals are known to be periodic over the 16384 samples, the process is valid.

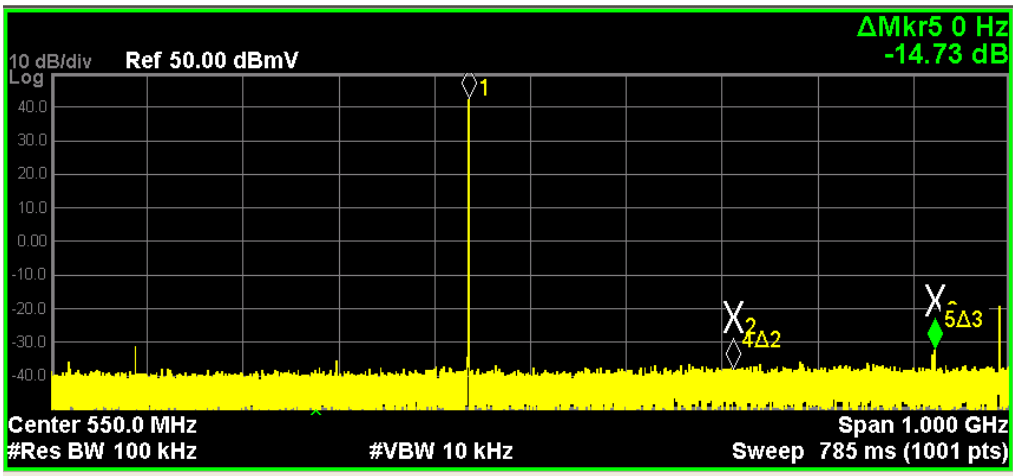
The measured results are detailed in Table 4.5. The new measurements are taken at close

Table 4.5: Attenuation of spurs with Pre-distortion

$x[n]$ Frequency (MHz)	$x_{2H}(t)$ Attenuation (dB)	$x_{F_s/2-F_c}(t)$ Attenuation (dB)
250	-13.3	-15.0
300	-15.2	-15.2
350	-14.3	-14.5
400	-6.8	-14.8
440	-16.5	-14.5
486	-14.7	-15.8
550	NA	-15.2
650	NA	-14.5
750	-14.0	-14.0
800	-17.4	-15.6
825	-18.2	-20.1
850	-18	-15.1
900	-19.7	-18.0
950	-19.4	-19.0
1000	-19.8	-19.0

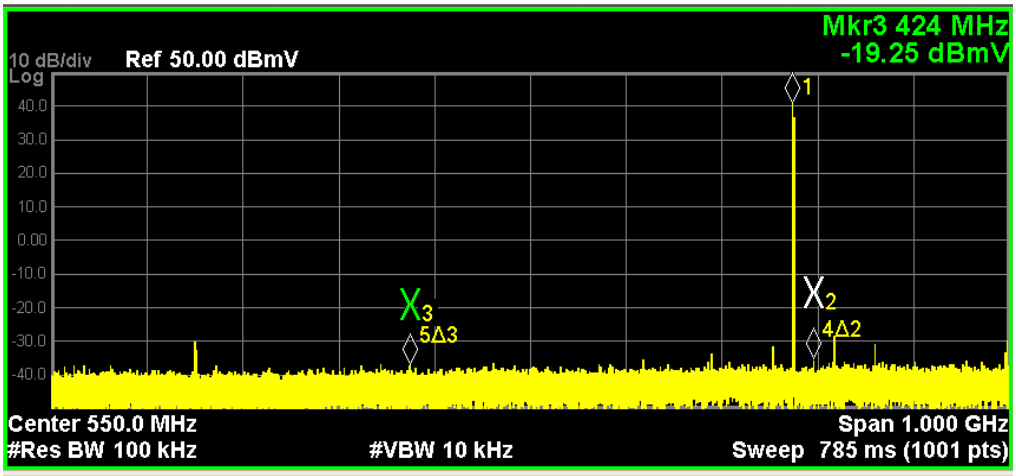
to 50MHz intervals instead of the initial data points to test the curve fitting ability of the algorithm.

Figure 4.15 has some screen captures of the DRFI band showing the effectiveness of the pre-distortion algorithm when sinusoids are used as the signal source. To show the effects of the pre-distortion a marker is set on each of the distortion signals, the marker delta function was used and then the pre-distortion was enabled. The numbers below the main display show the difference between the spur level for a normal and then a pre-distorted signal for each spur.



MKR	MODE	TRC	SCL	X	Y	FUNCTION	FUNCTION WIDTH	FUNCTION VALUE
1	N	1	f	486 MHz	42.13 dBmV			
2	F	1	f	762 MHz	-22.55 dBmV			
3	F	1	f	972 MHz	-17.74 dBmV			
4	Δ2	1	f	(Δ) 0 Hz	(Δ) -15.81 dB			
5	Δ3	1	f	(Δ) 0 Hz	(Δ) -14.73 dB			
6								
7								
8								
9								
10								
11								
12								

(a) $F_c = 486MHz$



MKR	MODE	TRC	SCL	X	Y	FUNCTION	FUNCTION WIDTH	FUNCTION VALUE
1	N	1	f	824 MHz	40.36 dBmV			
2	F	1	f	846 MHz	-15.65 dBmV			
3	F	1	f	424 MHz	-19.25 dBmV			
4	Δ2	1	f	(Δ) 0 Hz	(Δ) -20.09 dB			
5	Δ3	1	f	(Δ) 0 Hz	(Δ) -18.17 dB			
6								
7								
8								
9								
10								
11								
12								

(b) $F_c = 825MHz$

Figure 4.15: Pre-distorted sinusoids showing spur attenuation

4.4.1 Analysis of Results

The good results verify that the that the delays and filter coefficients are being used correctly. The pre-distortion signals are cancelling the spurs and no other distortion products are being created.

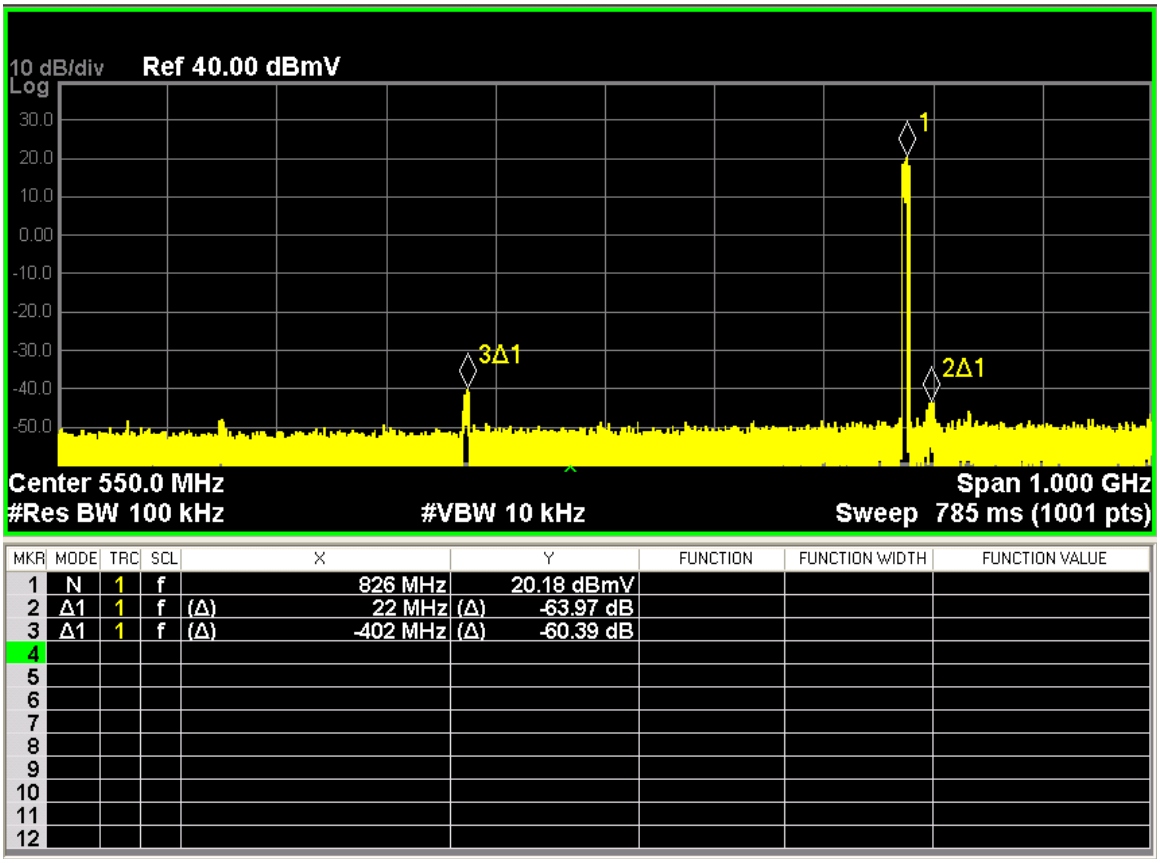
The distortion at $x_{F_s/2-F_c}(t)$ is attenuated by more than 14 dB across the band, which should be enough to meet the specification assuming there is no large drop in performance when a modulated signal is used. The $x_{2H}(t)$ distortion is attenuated by at least 14dB when the expected signal is above $747MHz$. This zone is the most critical because the spurs are more powerful and also because the DRFI specification for broadband noise must be met for aliased harmonics. The broadband noise specification is usually about 3dB more stringent than normal non-aliased harmonics.

4.4.2 Modification of code to work on an arbitrary input sample stream

Testing with sinusoids is of course very useful, but in order to be effective the pre-distortion structure must work on an arbitrary input signal. The code is changed to work in a manner that much more closely approximates a real time digital system.

The first trial used a subroutine to generate the desired signal and then two other subroutines are called that each create 16384 samples of a distortion signal and then the 3 sets of samples are added together. The new version has the distortion signal filtering subroutines incorporated into a loop that adds the desired signal with the 2 filter outputs for each sample. The time delay registers for both filters are updated simultaneously and the code operates much more closely to the way a DSP block would. The main differences are that the length of the file that is being pre-distorted is not fixed and the distortion samples are added to the expected sample at each iteration. The first test case of an arbitrary signal is a sinusoid generated by the DPG.

Using expected signal test frequencies that are similar to Tables 4.4 and 4.5. The distortion was verified to be attenuated by the same level. The performance is close enough to



MKR	MODE	TRC	SCL	X	Y	FUNCTION	FUNCTION WIDTH	FUNCTION VALUE
1	N	1	f	826 MHz	20.18 dBmV			
2	Δ1	1	f	(Δ) 22 MHz	(Δ) -63.97 dB			
3	Δ1	1	f	(Δ) -402 MHz	(Δ) -60.39 dB			
4								
5								
6								
7								
8								
9								
10								
11								
12								

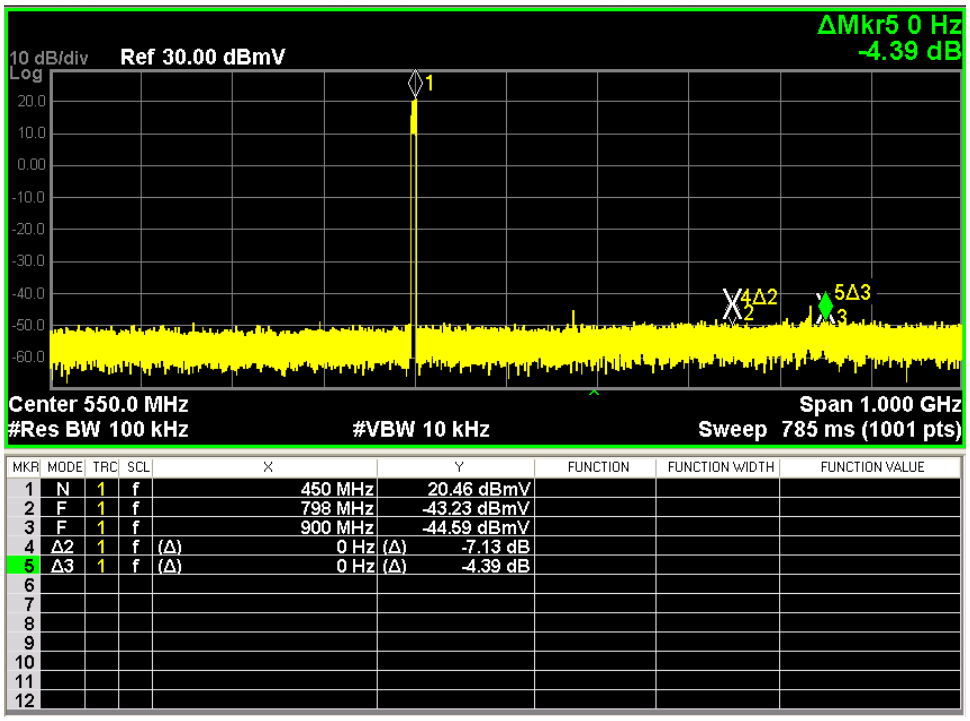
Figure 4.16: Screen Capture of Normal 825MHz QAM channel

Figure 4.15 that another screen capture is not shown. It is interesting to note there is a slight difference in the actual output frequency when the desired input is set to 850MHz. The CW generator developed for this investigation gives an actual output frequency of 824.64MHz while the DPG generated CW has an actual output of 824.94MHz. This is due to the differences in the algorithms used to generate a periodic sample loop.

While real modulated test signals can not usually be exported from the DPG as sample streams, 3 modulated signals formatted for the DPG were provided by Analog Devices to assist with this research. The first 2 files are samples for DOCSIS QAM signals centered at 450MHz and 825MHz. The 3rd is a sample file with the two signals combined. These sample files are 954368 samples long. Due to the nature of the function written there were no changes needed to use files of this size.

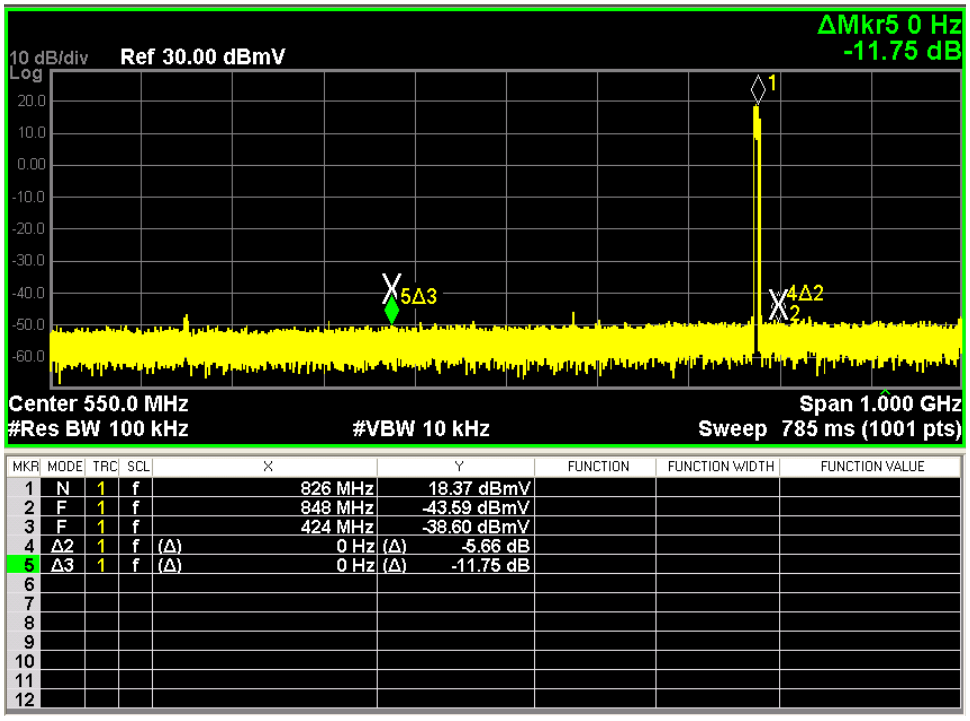
Figure 4.16 shows the normal QAM output of the DAC without pre-distortion. Marker 2 shows the presence of the aliased $x_{2H}(t)$ distortion and marker 3 shows the $x_{F_s/2-F_c}(t)$ distortion. Using marker points is not the most accurate way to make a measurement, but the marker delta function can be used to approximate that the spurious signals are at -64dBc and -60dBc respectively. The DRFI spec for these signals would be -64dBc [2]. Having a spurious emission right at the spec is not acceptable from a production standpoint, especially once temperature and unit to unit variations are factored in. Wide-band screen captures of the 2 pre-distorted individual QAM signals are shown in Figure 4.17.

[2]Specification according to DRFI I12 including the current 3dB relaxation for desired signals over 600MHz



MKR	MODE	TRC	SCL	X	Y	FUNCTION	FUNCTION WIDTH	FUNCTION VALUE
1	N	1	f	450 MHz	20.46 dBmV			
2	F	1	f	798 MHz	-43.23 dBmV			
3	F	1	f	900 MHz	-44.59 dBmV			
4	Δ2	1	f	(Δ) 0 Hz	(Δ) -7.13 dB			
5	Δ3	1	f	(Δ) 0 Hz	(Δ) -4.39 dB			
6								
7								
8								
9								
10								
11								
12								

(a) QAM at 450MHz



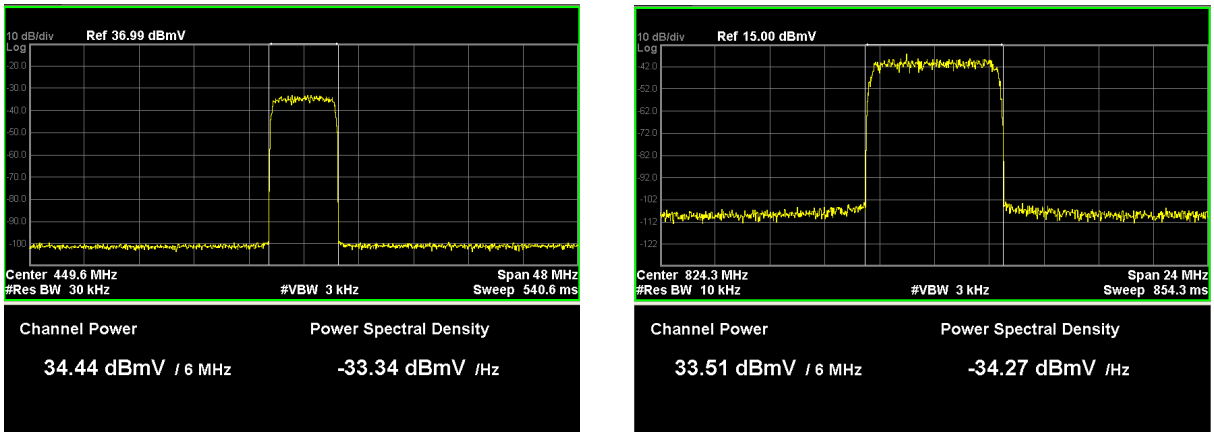
MKR	MODE	TRC	SCL	X	Y	FUNCTION	FUNCTION WIDTH	FUNCTION VALUE
1	N	1	f	826 MHz	18.37 dBmV			
2	F	1	f	848 MHz	-43.59 dBmV			
3	F	1	f	424 MHz	-38.60 dBmV			
4	Δ2	1	f	(Δ) 0 Hz	(Δ) -5.66 dB			
5	Δ3	1	f	(Δ) 0 Hz	(Δ) -11.75 dB			
6								
7								
8								
9								
10								
11								
12								

(b) QAM at 825MHz

Figure 4.17: Pre-distorted QAM showing spurious attenuation

The markers show the approximate attenuation of the spurious signals with pre-distortion. The spurious signals are clearly attenuated into the noise floor of the analyzer making an accurate measurement difficult without narrowing the band.

To make an accurate dBc measurement of the spurious signals, the power of the 2 QAMs are first measured using a narrower bandwidth. The power is found to be 34.44dBmV for the 450MHz QAM and 33.51dBmV for the 825MHz QAM. The power measurements are shown in Figure 4.18



(a) QAM at 450MHz

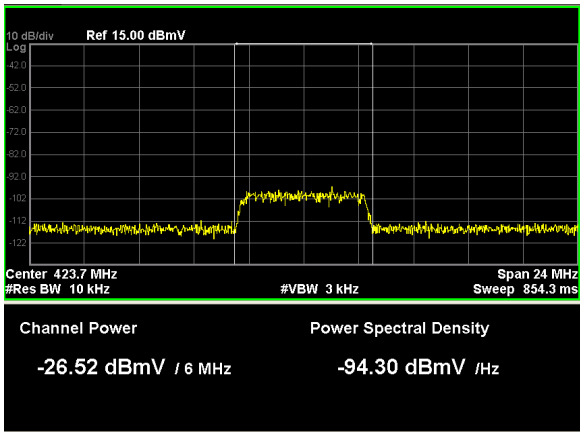
(b) QAM at 825MHz

Figure 4.18: QAM Channel Power

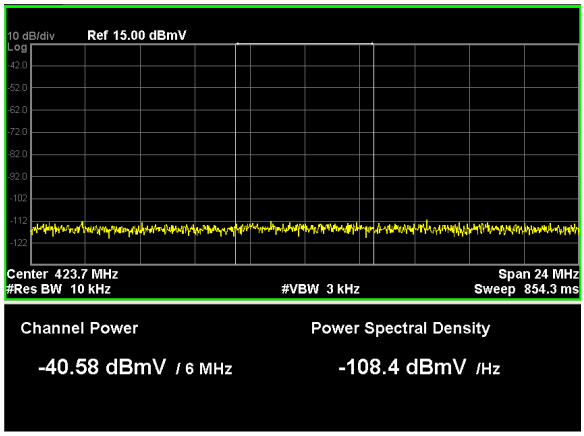
The spurs are measured in a 6MHz channel relative to the power of the carrier. Looking at the 825MHz QAM channel distortion first, Figures 4.19a and 4.19b show the reduction of the $x_{F_s/2-F_c}(t)$ distortion from -26.52dBmV or -60.03dBc to -40.58dBmV or -74.09dBc easily meeting the spec of -64dBc. The reduction of $x_{2H}(t)$ distortion is from -30.95dBmV or -64.46dBc to -37.49 or -71.00dBc. Using pre-distortion the system is meeting specification with 6dB margin.

Notice how the spectrum of the harmonic content is spread out over twice the bandwidth of the desired signal. This is caused by the squaring operation, which spreads the energy of the harmonic content over twice the bandwidth.

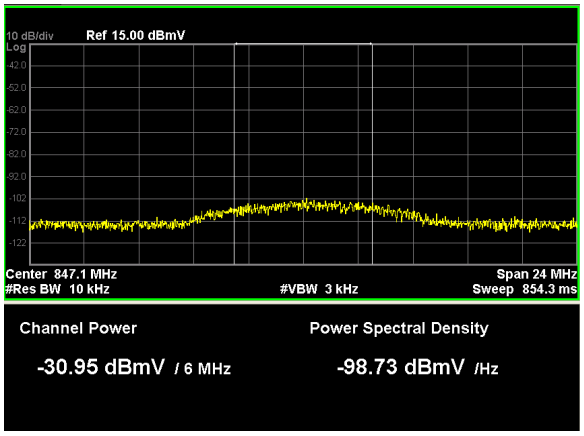
Channel power measurements of the spurs with the QAM output at 450MHz are shown in Figure 4.20. The distortion at $x_{F_s/2-F_c}(t)$ is reduced from -30.05dBmV or -64.49dBc,



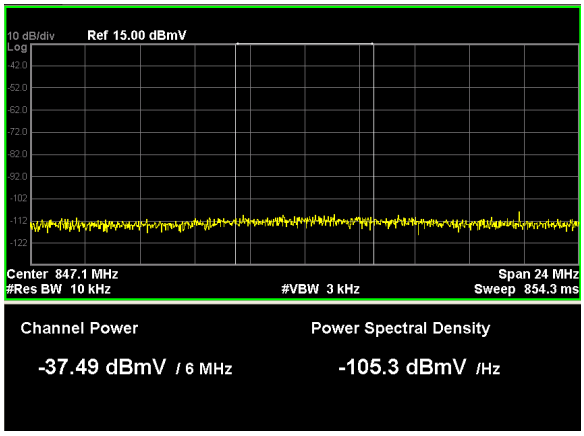
(a) $x_{F_s/2-F_c}[n]$ normal



(b) $x_{F_s/2-F_c}[n]$ Pre-distortion



(c) $x_{2H}[n]$ normal

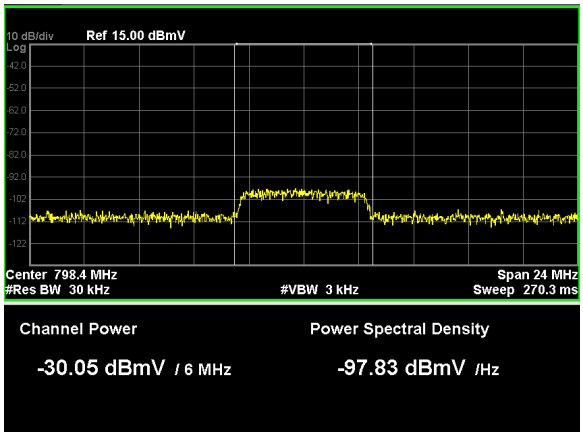


(d) $x_{2H}[n]$ Pre-distortion

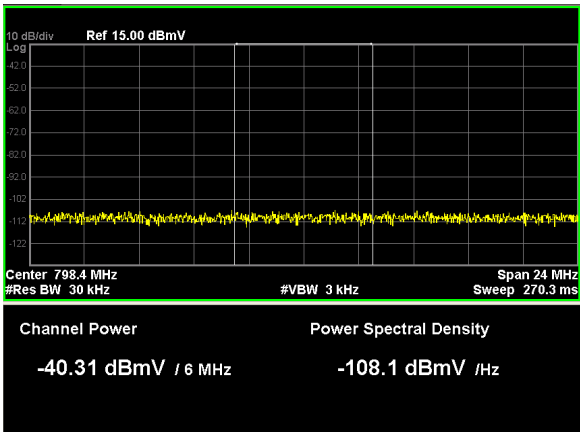
Figure 4.19: Normal and Pre-distored spurious measurements for QAM at 825MHz

just barely meeting spec, to -40.31dBmV or -74.74dBc which easily meets spec. The $x_{2H}(t)$ distortion was reduced from -33.08dBmV or -65.39dBc to -36.00dBmV or -70.44dBc. The harmonic content is not reduced quite as much but since the normal non-aliased 2^{nd} harmonic spec is -63dBc instead of -64dBc the margin is slightly better.

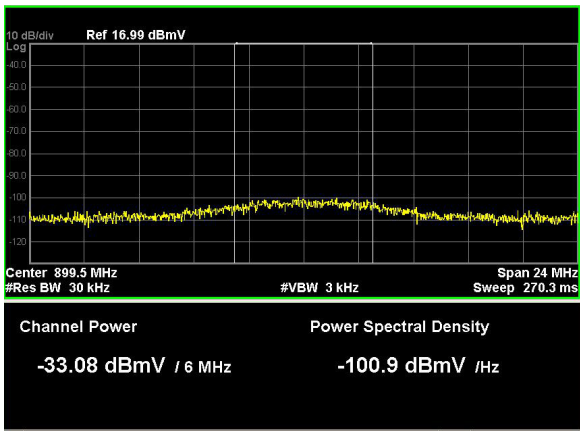
Finally the pre-distortion routine was tested with the combined signal containing both QAM's. The results with and without pre-distortion are in Figure 4.21. All four spurious signals are attenuated to the point where they easily meet spec and there are no extra distortion products added. The markers show an approximation of the reduction in spur power.



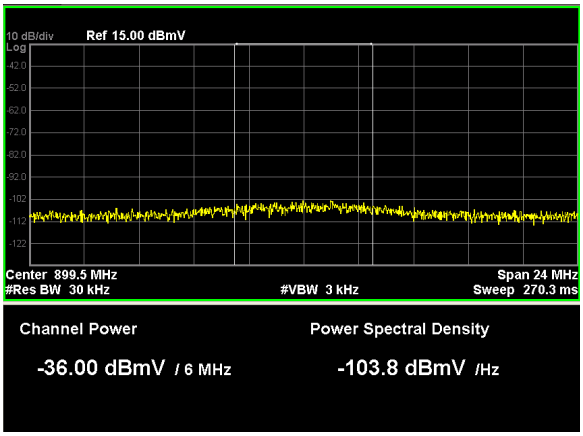
(a) $x_{F_s/2-F_c}(t)$ Normal



(b) $x_{F_s/2-F_c}(t)$ Pre-distortion

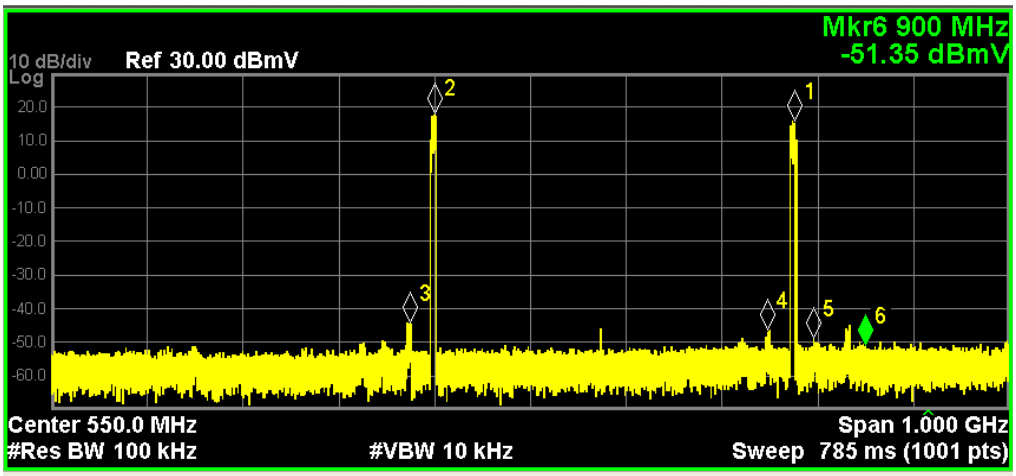


(c) $x_{2H}(t)$ Normal



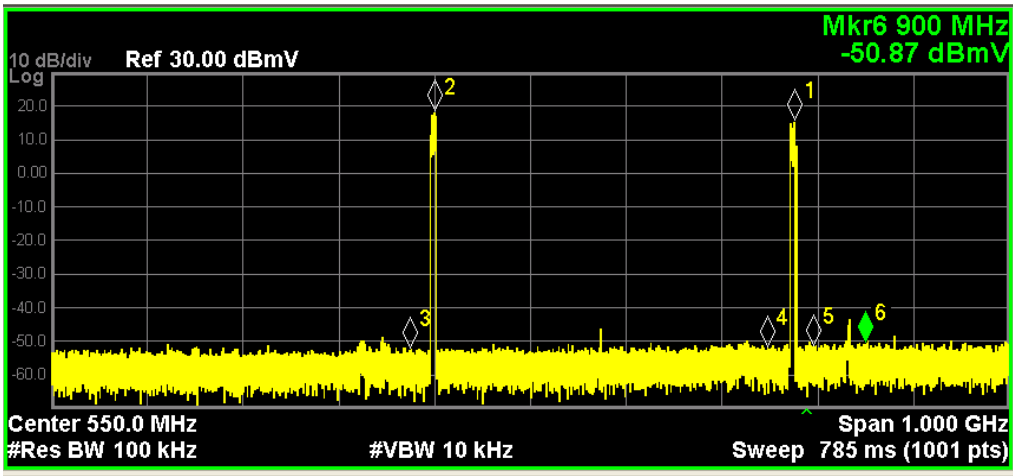
(d) $x_{2H}(t)$ Pre-distortion

Figure 4.20: Normal and Pre-distored spurious measurements for QAM at 450MHz



MKR	MODE	TRC	SCL	X	Y	FUNCTION	FUNCTION WIDTH	FUNCTION VALUE
1	N	1	f	826 MHz	15.26 dBmV			
2	N	1	f	450 MHz	17.59 dBmV			
3	N	1	f	400 MHz	-44.56 dBmV			
4	N	1	f (Δ)	800 MHz	(Δ) -46.92 dBmV			
5	N	1	f (Δ)	846 MHz	(Δ) -49.22 dBmV			
6	N	1	f	900 MHz	-51.35 dBmV			
7								
8								
9								
10								
11								
12								

(a) $Un - distorted$



MKR	MODE	TRC	SCL	X	Y	FUNCTION	FUNCTION WIDTH	FUNCTION VALUE
1	N	1	f	826 MHz	15.28 dBmV			
2	N	1	f	450 MHz	17.99 dBmV			
3	N	1	f	400 MHz	-52.57 dBmV			
4	N	1	f (Δ)	800 MHz	(Δ) -52.00 dBmV			
5	N	1	f (Δ)	846 MHz	(Δ) -51.78 dBmV			
6	N	1	f	900 MHz	-50.87 dBmV			
7								
8								
9								
10								
11								
12								

(b) $Pre - Distortion$

Figure 4.21: 2QAM channels at $450MHz$ and $825MHz$

4.4.3 Design considerations and Comparison to Current Solutions

In order for a filter of this type to be implemented some other analyses would have to be performed. Due to the high sampling rate used by these DACs, data is transferred to them using several parallel data lines. While it has been verified that FIR filters can be modified to operate in a parallel data structure a similar structure would still have developed and proven to work with an IIR filter.

It would be interesting to compare the resource requirements for the vendor supplied pre-distortion routines with the one developed here. Unfortunately due to their proprietary and confidential nature, the vendor algorithms can not be used for a comparison as to what method is more resource intensive. Both the vendor algorithms and the one presented in this thesis require some calibration for pre-distortion to be effective. It is assumed that this calibration would have to be performed on a unit to unit basis for both methods.

5. Conclusion

A method of measuring distortion products in an RFDAC and the system developed to cancel out these products has been outlined. Most pre-distortion techniques found in literature involve modelling the internal structure of an RFDAC, estimating the parameters of the model and then compensating for the things that result in distortion. The method detailed in this thesis assumes no knowledge of the internal structure of the RFDAC but uses digital filtering techniques to approximate the measured frequency response of the distortion products and add signals that cancel out the distortion.

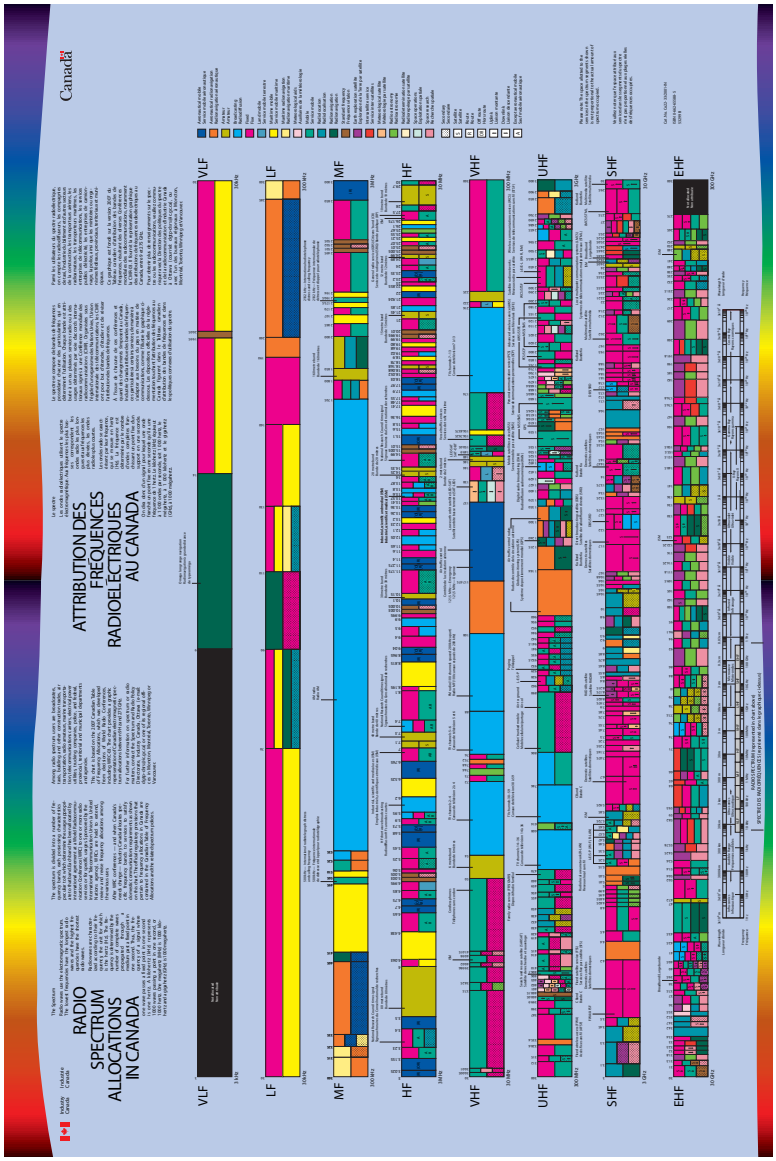
The main goal of developing a pre-distortion algorithm that improves the output of an RFDAC to a point where it is DRFI compliant has been achieved. Two different methods of matching the frequency response of the distortion products have been developed, one that is FIR filter based and one that is IIR filter based. The IIR filters were used to test out the pre-distortion algorithm on 3 different signals that could be possible outputs in a DOCSIS compliant system. In all cases the output of the DAC is DRFI compliant with significant margin.

While the specification is met there is still an issue where there is a significant difference between the measured results with a CW and a modulated signal. There could be several reasons for this including 2nd harmonic content caused by a higher order polynomial, such as 4th order products but this is difficult to determine given the test signals available. Most likely if the original measurement could be made with a modulated signal instead of a CW the measured frequency response would be closer to what is needed to cancel the harmonics of the modulated signal.

Further work will have to be done to integrate the pre-distortion routine into a real sys-

tem. This would require detailed knowledge of the actual system making it implementation specific and involve proprietary information. Therefore such work is beyond the scope of this thesis.

A. Radio Spectrum Allocations in Canada



References

[1] D. Large and J. Farmer, *Broadband Cable Access Networks*. Morgan Kaufmann, 2009.

[2] R. Horak, *Telecommunications and Data Communications Handbook*. Wiley, 2008. [Online]. Available: http://books.google.ca/books?id=rQ9bPgAACAAJ

[3] S. Ovadia, *Broadband Cable TV Access Networks*. Prentice Hall, 2001.

[4] L. D'Luna, L. Tan, D. Mueller, J. Laskowski, K. Cameron, J.-Y. Lee, D. Gee, J. Monroe, H. Law, J. Chang, M. Wakayama, T. Kwan, C.-H. Lin, A. Buchwald, T. Kaylani, F. Lu, T. Spieker, R. Hawley, and H. Smaueli, "A single-chip universal cable set-top box/modem transceiver," *Solid-State Circuits, IEEE Journal of*, vol. 34, no. 11, pp. 1647–1660, 1999.

[5] CableLabs, *Data Over Cable Service Interface Specification DOCSIS 3.0-Physical Layer Specification*. CM-SP-PHYv3.0-I03-070223, 2007.

[6] NCTA webmaster. (2013, Aug.) National cable and telecommunications association. . [Online]. Available: http://www.ncta.com/industry-data

[7] CableLabs, *Data Over Cable Service Interface Specification Downstream RF Interface Specification*. CM-SP-DRFI-I12-111117, 2011.

[8] M. Windisch and G. Fettweis, "Adaptive i/q imbalance compensation in low-if transmitter architectures," in *Vehicular Technology Conference, 2004. VTC2004-Fall. 2004 IEEE 60th*, vol. 3, 2004, pp. 2096–2100 Vol. 3.

[9] N. Zimmermann, B. Thiel, R. Negra, and S. Heinen, "System architecture of an rf-dac based multistandard transmitter," in *Circuits and Systems, 2009. MWSCAS '09. 52nd IEEE International Midwest Symposium on*, 2009, pp. 248–251.

[10] C. E. Shannon, "A mathematical theory of communication," *The Bell System Technical Journal*, vol. 27, pp. 379–423, 623–656, july, october 1948. [Online]. Available: http://www3.alcatel-lucent.com/bstj/vol27-1948/articles/bstj27-3-379.pdf

[11] S. Luschas, R. Schreier, and H.-S. Lee, "Radio frequency digital-to-analog converter," *Solid-State Circuits, IEEE Journal of*, vol. 39, no. 9, pp. 1462–1467, 2004.

[12] G. Engel, D. Fague, and A. Toledano, "Rf digital-to-analog converters enable direct synthesis of communications signals," *Communications Magazine, IEEE*, vol. 50, no. 10, pp. 108–116, 2012.

[13] Xilinx. (2013, Aug.) Virtex-7 t and xt fpgas data sheet. pdf. [Online]. Available: http://www.xilinx.com/support/documentation/data_sheets/ds183-Virtex_7_Data_Sheet.pdf

[14] M. Alavi, G. Voicu, R. Staszewski, L. de Vreede, and J. Long, "A 2×13-bit all-digital i/q rf-dac in 65-nm cmos," in *Radio Frequency Integrated Circuits Symposium (RFIC), 2013 IEEE*, 2013, pp. 167–170.

[15] Douglas A. Mercer. (2013, Aug.) Digital-to-Analog Converter Design. pdf. [Online]. Available: http://www.ecse.rpi.edu/homepages/ieee/presentations/RPI_DAC_Lecture_Oct_08.pdf

[16] M. Park, M. Perrott, and R. Staszewski, "An amplitude resolution improvement of an rf-dac employing pulsewidth modulation," *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol. 58, no. 11, pp. 2590–2603, 2011.

[17] N. Naskas and Y. Papananos, "A convergence-free predistortion technique for adaptive linearization of rf power amplifiers," *Analog Integrated Circuite and Signal Processing*, vol. 41, no. 2-3, pp. 109–118, Dec. 2004.

[18] G. Kardaras, L. Dittmann, and J. Soler, "Simplified polynomial digital predistortion for multimode software defined radios," in *Applied Sciences in Biomedical and Communication Technologies (ISABEL), 2010 3rd International Symposium on*, 2010, pp. 1–5.

[19] D. Johns and K. Martin, *Analog Integrated Circuit Design*, ser. Analog Integrated Circuit Design. Wiley, 1997. [Online]. Available: http://books.google.ca/books?id=1OIJZzLvVhcC

[20] W. Kester and Analog Devices Inc, *The data conversion handbook*, ser. Analog Devices Series. Butterworth-Heinemann Limited, 2005. [Online]. Available: http://books.google.ca/books?id=0aeBS6SgtR4C

[21] D. Dennis and G. Christopher, "Digital to analog converter," U.S. Patent 5 967 657, 1999.

[22] M.-J. Choe, Kwang-Hyun-Baek, and M. Teshome, "A 1.6gs/s 12b return-to-zero gaas rf dac for multiple nyquist operation," in *Solid-State Circuits Conference, 2005. Digest of Technical Papers. ISSCC. 2005 IEEE International*, 2005, pp. 112–587 Vol. 1.

[23] Analog Devices Inc. (2013, Aug.) Digital-to-Analog Converters AD9737A/AD9739A. pdf. [Online]. Available: http://www.analog.com/static/imported-files/data_sheets/AD9739.pdf

[24] S. Park, G. Kim, S.-C. Park, and W. Kim, "A digital-to-analog converter based on differential-quad switching," *Solid-State Circuits, IEEE Journal of*, vol. 37, no. 10, pp. 1335–1338, 2002.

[25] G. Engel, S. Kuo, and S. Rose, "A 14b 3/6ghz current-steering rf dac in 0.18 μm cmos with 66db aclr at 2.9ghz," in *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2012 IEEE International*, 2012, pp. 458–460.

[26] Agilent Technologies Inc. (2006) Application note 150, Agilent Spectrum Analysis Basics.

[27] B. Mohr, W. Li, and S. Heinen, "Analysis of digital predistortion architectures for direct digital-to-rf transmitter systems," in *Circuits and Systems (MWSCAS), 2012 IEEE 55th International Midwest Symposium on*, 2012, pp. 650–653.

[28] M. Lang, D. ing Mathias Lang, and I. Juni, "Algorithms for the constrained design of digital filters with arbitrary magnitude and phase responses," Ph.D. dissertation, Vienna Technical University of Technology, Vienna Austria, 1999.